

Beyond Linux® From Scratch (systemd Edition)

Version 12.2

The BLFS Development Team

Copyright © 1999-2024 The BLFS Development Team

Copyright © 1999-2024, The BLFS Development Team

All rights reserved.

This book is licensed under a [Creative Commons License](#).

Computer instructions may be extracted from the book under the [MIT License](#).

Linux® is a registered trademark of Linus Torvalds.

Published 2024-04-01

Revision History

Revision 12.2	2024-09-01	Thirtieth Release
Revision 12.1	2024-03-01	Twenty-ninth Release
Revision 12.0	2023-09-01	Twenty-eighth Release
Revision 11.3	2023-03-01	Twenty-seventh Release
Revision 11.2	2022-09-01	Twenty-sixth Release
Revision 11.1	2022-03-01	Twenty-fifth Release
Revision 11.0	2021-09-01	Twenty-fourth Release
Revision 10.1	2021-03-01	Twenty-third Release
Revision 10.0	2020-09-01	Twenty-second Release
Revision 9.1	2020-03-01	Twenty-first Release
Revision 9.0	2019-09-01	Twentieth release
Revision 8.4	2019-03-01	Nineteenth release
Revision 8.3	2018-09-01	Eighteenth release
Revision 8.2	2018-03-02	Seventeenth release
Revision 8.1	2017-09-01	Sixteenth release
Revision 8.0	2017-02-25	Fifteenth release
Revision 7.10	2016-09-07	Fourteenth release
Revision 7.9	2016-03-08	Thirteenth release
Revision 7.8	2015-10-01	Twelfth release
Revision 7.7	2015-03-06	Eleventh release
Revision 7.6	2014-09-23	Tenth release
Revision 7.5	2014-03-05	Ninth release
Revision 7.4	2013-09-14	Eighth release
Revision 6.3	2008-08-24	Seventh release
Revision 6.2	2007-02-14	Sixth release
Revision 6.1	2005-08-14	Fifth release
Revision 6.0	2005-04-02	Fourth release
Revision 5.1	2004-06-05	Third release
Revision 5.0	2003-11-06	Second release
Revision 1.0	2003-04-25	First release

Abstract

This book follows on from the Linux From Scratch book. It introduces and guides the reader through additions to the system including networking, graphical interfaces, sound support, and printer and scanner support.

Dedication

This book is dedicated to the LFS community

Table of Contents

[Preface](#)

[Foreword](#)

[Who Would Want to Read this Book](#)

[Organization](#)

I. Introduction

1. Welcome to BLFS

[Which Sections of the Book Do I Want?](#)
[Conventions Used in this Book](#)
[Book Version](#)
[Mirror Sites](#)
[Getting the Source Packages](#)
[Change Log](#)
[Mailing Lists](#)
[Editor Notes](#)
[Asking for Help and the FAQ](#)
[Credits](#)
[Contact Information](#)

2. Important Information

[Notes on Building Software](#)
[The /usr Versus /usr/local Debate](#)
[Optional Patches](#)
[BLFS Systemd Units](#)
[About Libtool Archive \(.la\) files](#)
[Libraries: Static or shared?](#)
[Locale Related Issues](#)
[Going Beyond BLFS](#)

II. Post LFS Configuration and Extra Software

3. After LFS Configuration Issues

[Creating a Custom Boot Device](#)
[About Console Fonts](#)
[About Firmware](#)
[About Devices](#)
[Configuring for Adding Users](#)
[About System Users and Groups](#)
[The Bash Shell Startup Files](#)
[The /etc/vimrc and ~/.vimrc Files](#)
[Customizing your Logon with /etc/issue](#)

4. Security

[Vulnerabilities](#)
[make-ca-1.14](#)
[CrackLib-2.10.2](#)
[cryptsetup-2.7.4](#)
[Cyrus SASL-2.1.28](#)
[GnuPG-2.4.5](#)
[GnuTLS-3.8.7.1](#)
[GPGME-1.23.2](#)
[iptables-1.8.10](#)
[Setting Up a Network Firewall](#)
[libcap-2.70 with PAM](#)
[Linux-PAM-1.6.1](#)
[liboauth-1.0.3](#)
[libpwquality-1.4.5](#)
[MIT Kerberos V5-1.21.3](#)
[Nettle-3.10](#)
[NSS-3.103](#)
[OpenSSH-9.8p1](#)
[p11-kit-0.25.5](#)
[Polkit-125](#)
[polkit-gnome-0.105](#)
[Shadow-4.16.0](#)
[ssh-askpass-9.8p1](#)
[stunnel-5.72](#)
[Sudo-1.9.15p5](#)
[Tripwire-2.4.3.7](#)

5. File Systems and Disk Management

[About initramfs](#)
[btrfs-progs-6.10.1](#)
[dosfstools-4.2](#)
[Fuse-3.16.2](#)
[jfsutils-1.1.15](#)
[LVM2-2.03.26](#)
[About Logical Volume Management \(LVM\)](#)
[About RAID](#)

[mdadm-4.3](#)
[ntfs-3g-2022.10.3](#)
[gptfdisk-1.0.10](#)
[parted-3.6](#)
[smartmontools-7.4](#)
[sshfs-3.7.3](#)
[xfsprogs-6.9.0](#)

Packages for UEFI Boot

[efivar-39](#)
[efibootmgr-18](#)
[GRUB-2.12 for EFI](#)
[Using GRUB to Set Up the Boot Process with UEFI](#)

6. Text Editors

[Bluefish-2.2.15](#)
[Ed-1.20.2](#)
[Emacs-29.4](#)
[Gedit-47.0](#)
[JOE-4.6](#)
[kate-24.08.0](#)
[Mousepad-0.6.2](#)
[Nano-8.1](#)
[Vim-9.1.0660](#)

7. Shells

[Dash-0.5.12](#)
[Tcsh-6.24.13](#)
[zsh-5.9](#)

8. Virtualization

[qemu-9.0.2](#)

III. General Libraries and Utilities

9. General Libraries

[Abseil-cpp-20240722.0](#)
[AppStream-1.0.3](#)
[appstream-glib-0.8.3](#)
[Apr-1.7.4](#)
[Apr-Util-1.6.3](#)
[Aspell-0.60.8.1](#)
[boost-1.86.0](#)
[brotli-1.1.0](#)
[CLucene-2.3.3.4](#)
[dbus-glib-0.112](#)
[Double-conversion-3.3.0](#)
[duktape-2.7.0](#)
[enchant-2.8.2](#)
[Exempi-2.6.5](#)
[fftw-3.3.10](#)
[fmt-11.0.2](#)
[GLib-2.80.4](#)
[GLibmm-2.66.7](#)
[GLibmm-2.80.0](#)
[GMime-3.2.7](#)
[gsl-2.8](#)
[gspell-1.12.2](#)
[highway-1.2.0](#)
[icu-75.1](#)
[inih-58](#)
[Intel-gmmlib-22.4.1](#)
[Jansson-2.14](#)
[JSON-C-0.17](#)
[JSON-GLib-1.8.0](#)
[keyutils-1.6.3](#)
[libaio-0.3.113](#)
[libarchive-3.7.4](#)
[libassuan-3.0.1](#)
[libatasmart-0.19](#)
[libatomic_ops-7.8.2](#)
[libblockdev-3.1.1](#)
[libbytesize-2.11](#)
[libclc-18.1.7](#)
[libcloudproviders-0.3.6](#)
[libdaemon-0.14](#)
[libdisplay-info-0.2.0](#)

[libgcrypt-1.11.0](#)
[libgpg-error-1.50](#)
[libgsf-1.14.52](#)
[libgudev-238](#)
[libgusb-0.4.9](#)
[libical-3.0.18](#)
[libidn-1.42](#)
[libidn2-2.3.7](#)
[libksba-1.6.7](#)
[liblinear-247](#)
[libmbim-1.26.4](#)
[libnvme-1.10](#)
[libpaper-2.2.5](#)
[libportal-0.7.1](#)
[libptytty-2.0](#)
[libqalculate-5.2.0](#)
[libqmi-1.30.8](#)
[libseccomp-2.5.5](#)
[libsigc++-2.12.1](#)
[libsigc++-3.6.0](#)
[libsigsegv-2.14](#)
[libssh2-1.11.0](#)
[libstatgrab-0.92.1](#)
[libtasn1-4.19.0](#)
[libunistring-1.2](#)
[libunwind-1.6.2](#)
[liburcu-0.14.0](#)
[libusb-1.0.27](#)
[libuv-1.48.0](#)
[libxkbcommon-1.7.0](#)
[libxml2-2.13.3](#)
[libxmlb-0.3.19](#)
[libxslt-1.1.42](#)
[libwacom-2.12.2](#)
[libyaml-0.2.5](#)
[log4cplus-2.1.1](#)
[LZO-2.10](#)
[mtdev-1.1.7](#)
[Node.js-20.16.0](#)
[npth-1.7](#)
[NSPR-4.35](#)
[PCRE2-10.44](#)
[Popt-1.19](#)
[Protobuf-27.3](#)
[Protobuf-c-1.5.0](#)
[Qca-2.3.9](#)
[qcoro-0.10.0](#)
[SpiderMonkey from firefox-115.14.0](#)
[SPIRV-Headers-1.3.290.0](#)
[SPIRV-Tools-1.3.290.0](#)
[SPIRV-LLVM-Translator-18.1.3](#)
[Talloc-2.4.2](#)
[Uchardet-0.0.8](#)
[Umockdev-0.18.3](#)
[utfcpp-4.0.5](#)
[Wayland-1.23.0](#)
[Wayland-Protocols-1.36](#)
[wv-1.2.9](#)
[Xapian-1.4.26](#)

10. Graphics and Font Libraries

[AAlib-1.4rc5](#)
[babl-0.1.108](#)
[Exiv2-0.28.3](#)
[FreeType-2.13.3](#)
[Fontconfig-2.15.0](#)
[FriBidi-1.0.15](#)
[gegl-0.4.48](#)
[giflib-5.2.2](#)
[Glad-2.0.6](#)
[GLM-1.0.1](#)
[Graphite2-1.3.14](#)
[harfBuzz-9.0.0](#)
[JasPer-4.2.4](#)
[Little CMS-2.16](#)
[libavif-1.1.1](#)

[libexif-0.6.24](#)
[libgxps-0.3.2](#)
[libjpeg-turbo-3.0.1](#)
[libjxl-0.10.3](#)
[libmng-2.0.3](#)
[libmypaint-1.6.1](#)
[libpng-1.6.43](#)
[libraw-0.21.2](#)
[librsvg-2.58.3](#)
[Libspiro-20220722](#)
[libtiff-4.6.0](#)
[libwebp-1.4.0](#)
[mypaint-brushes-1.3.1](#)
[newt-0.52.24](#)
[opencv-4.10.0](#)
[OpenJPEG-2.5.2](#)
[Pixman-0.43.4](#)
[Poppler-24.08.0](#)
[Potrace-1.16](#)
[Qpdf-11.9.1](#)
[qrencode-4.1.1](#)
[sassc-3.6.2](#)
[weebp-pixbuf-loader-0.2.7](#)
[woff2-1.0.2](#)

11. General Utilities

[Asciidoctor-2.0.23](#)
[Bogofilter-1.2.5](#)
[Compface-1.5.2](#)
[desktop-file-utils-0.27](#)
[dos2unix-7.5.2](#)
[Graphviz-12.1.0](#)
[GTK-Doc-1.34.0](#)
[Highlight-4.13](#)
[ibus-1.5.30](#)
[ImageMagick-7.1.1-36](#)
[ISO Codes-4.16.0](#)
[Isof-4.99.0](#)
[mandoc-1.14.6](#)
[pinentry-1.3.1](#)
[Screen-4.9.1](#)
[shared-mime-info-2.4](#)
[Sharutils-4.15.2](#)
[tidy-html5-5.8.0](#)
[Time-1.9](#)
[tree-2.1.3](#)
[unixODBC-2.3.12](#)
[xdg-dbus-proxy-0.1.5](#)
[Xdg-user-dirs-0.18](#)

12. System Utilities

[AccountsService-23.13.9](#)
[acpid-2.0.34](#)
[at-3.2.5](#)
[autofs-5.1.9](#)
[BlueZ-5.77](#)
[Bubblewrap-0.9.0](#)
[Colord-1.4.7](#)
[cpio-2.15](#)
[cups-pk-helper-0.2.7](#)
[dbus-1.14.10](#)
[Fcron-3.2.1](#)
[GPM-1.20.7](#)
[Hdparm-9.65](#)
[hwdata-0.385](#)
[LSB-Tools-0.12](#)
[Lm-sensors-3-6-0](#)
[Logrotate-3.22.0](#)
[MC-4.8.32](#)
[ModemManager-1.18.12](#)
[notification-daemon-3.20.0](#)
[p7zip-17.04](#)
[Pax-20240817](#)
[pciutils-3.13.0](#)
[pm-utils-1.4.1](#)
[Power-profiles-daemon-0.21](#)
[Raptor-2.0.16](#)

[Rasqal-0.9.33](#)
[Redland-1.0.17](#)
[sg3_utils-1.48](#)
[sysmon-qt-2.0.1](#)
[Sysstat-12.7.6](#)
[Systemd-256.4](#)
[UDisks-2.10.1](#)
[UnRAR-7.0.9](#)
[UnZip-6.0](#)
[UPower-1.90.4](#)
[usbutils-017](#)
[Which-2.21 and Alternatives](#)
[xdotool-3.20211022.1](#)
[Zip-3.0](#)

13. Programming

[Cbindgen-0.27.0](#)
[Clisp-2.49](#)
[CMake-3.30.2](#)
[Doxygen-1.12.0](#)
[dtc-1.7.1](#)
[GCC-14.2.0](#)
[GC-8.2.6](#)
[GDB-15.1](#)
[Git-2.46.0](#)
[Running a Git Server](#)
[Guile-3.0.10](#)
[LLVM-18.1.7](#)
[Lua-5.4.7](#)
[Lua-5.2.4](#)
[Mercurial-6.8.1](#)
[NASM-2.16.03](#)
[Patchelf-0.18.0](#)
[Perl Modules](#)
[Perl Module Dependencies](#)
[PHP-8.3.10](#)
[Python-3.12.5](#)
[Python-3.11.1](#)
[Python Modules](#)
[Python Dependencies](#)
[Ruby-3.3.4](#)
[Rustc-1.80.1](#)
[rust-bindgen-0.70.0](#)
[SCons-4.8.0](#)
[slang-2.3.3](#)
[Subversion-1.14.3](#)
[Running a Subversion Server](#)
[SWIG-4.2.1](#)
[Tk-8.6.14](#)
[unifdef-2.12](#)
[Vala-0.56.17](#)
[Valgrind-3.23.0](#)
[yasm-1.3.0](#)

Java

[Java-22.0.2](#)
[OpenJDK-22.0.2](#)
[Configuring the Java environment](#)
[apache-ant-1.10.14](#)

IV. Networking

14. Connecting to a Network

[Advanced Network Setup](#)
[dhcpcd-10.0.8](#)

15. Networking Programs

[bridge-utils-1.7.1](#)
[cifs-utils-7.0](#)
[NcFTP-3.2.7](#)
[Net-tools-2.10](#)
[NFS-Utils-2.6.4](#)
[ntp-4.2.8p18](#)
[rpcbind-1.2.7](#)
[rsync-3.3.0](#)
[Samba-4.20.4](#)
[Wget-1.24.5](#)

[Configuring the Linux Kernel for Wireless](#)
[iw-6.9](#)
[Wireless Tools-29](#)
[wpa_supplicant-2.11](#)

[16. Networking Utilities](#)

[Avahi-0.8](#)
[BIND Utilities-9.20.0](#)
[NetworkManager-1.48.8](#)
[network-manager-applet-1.34.0](#)
[Nmap-7.95](#)
[Traceroute-2.1.5](#)
[Whois-5.4.3](#)
[Wireshark-4.2.6](#)

[17. Networking Libraries](#)

[c-ares-1.33.0](#)
[cURL-8.9.1](#)
[GeoClue-2.7.1](#)
[glib-networking-2.80.0](#)
[kdsoap-2.2.0](#)
[Idns-1.8.4](#)
[libevent-2.1.12](#)
[libmnl-1.0.5](#)
[libndp-1.9](#)
[libnl-3.10.0](#)
[libnma-1.10.6](#)
[libslnl-2.0.1](#)
[libpcap-1.10.4](#)
[libpsl-0.21.5](#)
[libslirp-4.8.0](#)
[libsoup-2.74.3](#)
[libsoup-3.4.4](#)
[libtirpc-1.3.5](#)
[neon-0.33.0](#)
[nghttp2-1.62.1](#)
[rpcsvc-proto-1.4.4](#)
[Serf-1.3.10](#)
[uhtpmock-0.11.0](#)

[18. Text Web Browsers](#)

[Links-2.30](#)
[Lynx-2.9.2](#)

[19. Mail/News Clients](#)

[Fetchmail-6.4.39](#)
[mailx-12.5](#)
[Mutt-2.2.13](#)
[Procmail-3.24](#)
[Other Mail and News Programs](#)

V. Servers

[20. Major Servers](#)

[Apache-2.4.62](#)
[BIND-9.20.0](#)
[Kea 2.6.1 DHCP Server](#)
[ProFTPD-1.3.8b](#)

[21. Mail Server Software](#)

[Dovecot-2.3.21.1](#)
[Exim-4.98](#)
[Postfix-3.9.0](#)
[sendmail-8.18.1](#)

[22. Databases](#)

[Important Notes About Upgrading Database Server Software](#)
[Imdb-0.9.31](#)
[MariaDB-10.11.8](#)
[PostgreSQL-16.4](#)
[SQLite-3.46.1](#)

[23. Other Server Software](#)

[OpenLDAP-2.6.8](#)
[Unbound-1.21.0](#)

VI. Graphical Components

24. Graphical Environments

[Introduction to Xorg-7](#)
[util-macros-1.20.1](#)
[xorgproto-2024.1](#)
[libXau-1.0.11](#)
[libXdmcP-1.1.5](#)
[xcb-proto-1.17.0](#)
[libxcb-1.17.0](#)
[Xorg_Libraries](#)
[libxcvt-0.1.2](#)
[xcb-util-0.4.1](#)
[xcb-util-image-0.4.1](#)
[xcb-util-keysyms-0.4.1](#)
[xcb-util-renderutil-0.3.10](#)
[xcb-util-wm-0.4.2](#)
[xcb-util-cursor-0.1.4](#)
[Mesa-24.1.5](#)
[xbitmaps-1.1.3](#)
[Xorg_Applications](#)
[luit-20240102](#)
[xcursor-themes-1.0.7](#)
[Xorg_Fonts](#)
[XKeyboardConfig-2.42](#)
[Xwayland-24.1.2](#)
[Xorg_Server-21.1.13](#)
[Xorg_Input_Drivers](#)
[twm-1.0.12](#)
[xterm-393](#)
[xclock-1.1.1](#)
[xinit-1.4.2](#)
[Xorg-7 Testing and Configuration](#)
[Tuning Fontconfig](#)
[TTF and OTF fonts](#)
[Xorg_Legacy](#).

25. Graphical Environment Libraries

[Atkmm-2.28.4](#)
[Atkmm-2.36.3](#)
[at-spi2-core-2.52.0](#)
[Cairo-1.18.0](#)
[libcairomm-1.0 \(cairomm-1.14.5\)](#)
[libcairomm-1.16 \(cairomm-1.18.0\)](#)
[colord-gtk-0.3.1](#)
[FLTK-1.3.9](#)
[Freeglut-3.6.0](#)
[gdk-pixbuf-2.42.12](#)
[GLEW-2.2.0](#)
[Glslang-14.3.0](#)
[GLU-9.0.3](#)
[GOffice-0.10.57](#)
[Grantlee-5.3.1](#)
[Graphene-1.10.8](#)
[GTK+-3.24.43](#)
[GTK-4.14.5](#)
[Gtkmm-3.24.9](#)
[Gtkmm-4.14.0](#)
[gtk-vnc-1.3.1](#)
[gtksourceview-3.24.11](#)
[gtksourceview4-4.8.4](#)
[gtksourceview5-5.12.1](#)
[imlib2-1.12.3](#)
[kColorPicker-0.3.1](#)
[kImageAnnotator-0.7.1](#)
[keybinder-3.0-0.3.2](#)
[libadwaita-1.5.3](#)
[libei-1.3.0](#)
[libgedit-amtk-5.8.0](#)
[libgedit-gtksourceview-299.2.1](#)
[libhandy-1.8.3](#)
[Libdrm-2.4.122](#)
[libepoxy-1.5.10](#)
[libnotify-0.8.3](#)
[libxklavier-5.4](#)
[Pango-1.54.0](#)
[Pangomm-2.46.4](#)
[Pangomm-2.54.0](#)

[qt5-components-5.15.14](#)
[Qt-6.7.2](#)
[QtWebEngine-6.7.2](#)
[startup-notification-0.12](#)
[Vulkan-Headers-1.3.294](#)
[Vulkan-Loader-1.3.294](#)
[WebKitGTK-2.44.3](#)
[xdg-desktop-portal-1.18.2](#)
[xdg-desktop-portal-gtk-1.15.1](#)

26. Display Managers

[GDM-46.2](#)
[lightdm-1.32.0](#)
[sddm-0.21.0](#)

27. Window Managers

[Fluxbox-1.3.7](#)
[IceWM-3.6.0](#)
[openbox-3.6.1](#)

28. Icons

[adwaita-icon-theme-46.2](#)
[breeze-icons-6.5.0](#)
[gnome-icon-theme-3.12.0](#)
[gnome-icon-theme-extras-3.12.0](#)
[gnome-icon-theme-symbolic-3.12.0](#)
[gnome-themes-extra-3.28](#)
[hicolor-icon-theme-0.18](#)
[icon-naming-utils-0.8.90](#)
[lxde-icon-theme-0.5.1](#)
[oxygen-icons-6.0.0](#)

VII. KDE

29. Introduction to KDE

[KDE Preliminaries](#)
[extra-cmake-modules-6.5.0](#)
[Phonon-4.12.0](#)
[Phonon-backend-vlc-0.12.0](#)
[Polkit-Qt-0.200.0](#)
[plasma-wayland-protocols-1.13.0](#)

30. KDE Frameworks 6

[KDE Frameworks 6 Pre-installation Configuration](#)
[Building KDE Frameworks 6.5.0 \(KF6\)](#)

31. KDE Frameworks 6 Based Applications

[ark-24.08.0](#)
[dolphin-24.08.0](#)
[dolphin-plugins-24.08.0](#)
[kdenlive-24.08.0](#)
[khelpcenter-24.08.0](#)
[Konsole-24.08.0](#)
[konversation-24.08.0](#)
[libkexiv2-24.08.0](#)
[plasma-activities-6.1.4](#)
[okular-24.08.0](#)
[libkdcraw-24.08.0](#)
[gwenview-24.08.0](#)
[libkcddb-24.08.0](#)
[K3b-24.08.0](#)
[Further KDE packages](#)

32. KDE Plasma

[kirigami-addons-1.4.0](#)
[pulseaudio-qt-1.5.0](#)
[Building Plasma](#)

VIII. GNOME

33. GNOME Libraries and Desktop

Libraries

[Gcr-3.41.2](#)
[Gcr-4.3.0](#)
[gsettings-desktop-schemas-46.1](#)
[libsecret-0.21.4](#)
[rest-0.9.1](#)

[totem-pl-parser-3.26.6](#)
[VTE-0.76.4](#)
[yelp-xsl-42.1](#)
[geocode-glib-3.26.4](#)
[Gjs-1.80.2](#)
[gnome-autoar-0.4.4](#)
[gnome-desktop-44.1](#)
[gnome-menus-3.36.0](#)
[gnome-online-accounts-3.50.4](#)
[Grilo-0.3.16](#)
[libgdata-0.18.1](#)
[libgee-0.20.6](#)
[libgtop-2.41.3](#)
[libgweather-4.4.2](#)
[libpeas-1.36.0](#)
[libshumate-1.2.3](#)
[evolution-data-server-3.52.4](#)
[Tracker-3.7.3](#)
[Tracker-miners-3.7.3](#)
[GSound-1.0.3](#)
[xdg-desktop-portal-gnome-46.2](#)

GNOME Desktop Components

[DConf-0.40.0 / DConf-Editor-45.0.1](#)
[gnome-backgrounds-46.0](#)
[Gvfs-1.54.2](#)
[gexiv2-0.14.3](#)
[Nautilus-46.2](#)
[gnome-bluetooth-46.1](#)
[gnome-keyring-46.2](#)
[gnome-settings-daemon-46.0](#)
[Tecla-46.0](#)
[gnome-control-center-46.4](#)
[Mutter-46.4](#)
[gnome-shell-46.4](#)
[gnome-shell-extensions-46.2](#)
[gnome-session-46.0](#)
[gnome-tweaks-46.1](#)
[gnome-user-docs-46.4](#)
[Yelp-42.2](#)

34. GNOME Applications

[Baobab-46.0](#)
[Brasero-3.12.3](#)
[EOG-45.4](#)
[Evince-46.3.1](#)
[Evolution-3.52.4](#)
[File-Roller-44.3](#)
[gnome-calculator-46.1](#)
[gnome-color-manager-3.36.0](#)
[gnome-connections-46.0](#)
[gnome-disk-utility-46.0](#)
[gnome-logs-45.0](#)
[gnome-maps-46.11](#)
[gnome-nettool-42.0](#)
[gnome-power-manager-43.0](#)
[gnome-screenshot-41.0](#)
[gnome-system-monitor-46.0](#)
[gnome-terminal-3.52.2](#)
[gnome-weather-46.0](#)
[Gucharmap-15.1.5](#)
[Seahorse-43.0](#)
[Snapshot-46.3](#)

IX. Xfce

35. Xfce Desktop

[libxfce4util-4.18.2](#)
[Xfconf-4.18.3](#)
[libxfce4ui-4.18.6](#)
[Exo-4.18.0](#)
[Garcon-4.18.2](#)
[libwnck-43.0](#)
[xfce4-panel-4.18.6](#)
[thunar-4.18.11](#)
[thunar-volman-4.18.0](#)
[tumbler-4.18.2](#)

[xfce4-appfinder-4.18.1](#)
[xfce4-power-manager-4.18.4](#)
[xfce4-settings-4.18.6](#)
[Xfdesktop-4.18.1](#)
[Xfwm4-4.18.0](#)
[xfce4-session-4.18.4](#)

36. Xfce Applications

[Parole-4.18.1](#)
[xfce4-terminal-1.1.3](#)
[Xfburn-0.7.2](#)
[Ristretto-0.13.2](#)
[xfce4-dev-tools-4.18.1](#)
[xfce4-notifyd-0.9.6](#)
[xfce4-pulseaudio-plugin-0.4.8](#)

X. LXQt

37. LXQt Desktop

[LXQt Desktop Pre-Install Instructions](#)

KF5, Plasma, and Misc Dependencies

[kwindowsystem-6.5.0 for lxqt](#)
[kconfig-6.5.0 for lxqt](#)
[solid-6.5.0 for lxqt](#)
[kidletime-6.5.0 for lxqt](#)
[kwayland-6.1.4 for lxqt](#)
[libkscreen-6.1.4 for lxqt](#)
[layer-shell-qt-6.1.4 for lxqt](#)
[muparser-2.3.4](#)

LXQt Desktop Components

[lxqt-build-tools-2.0.0](#)
[libqtxdg-4.0.0](#)
[lxqt-menu-data-2.0.0](#)
[liblxqt-2.0.0](#)
[libsysstat-1.0.0](#)
[qtxdg-tools-4.0.0](#)
[libfm-extra-1.3.2](#)
[menu-cache-1.1.0](#)
[libfm-qt-2.0.2](#)
[lxqt-globalkeys-2.0.0](#)
[libdbusmenu-lxqt-0.1.0](#)
[lxqt-qtplugin-2.0.0](#)
[lxqt-panel-2.0.1](#)
[pcmanfm-qt-2.0.0](#)
[lxqt-powermanagement-2.0.0](#)
[lxqt-runner-2.0.0](#)
[lxqt-themes-2.0.0](#)
[lxqt-admin-2.0.0](#)
[lxqt-about-2.0.0](#)
[lxqt-config-2.0.0](#)
[lxqt-policykit-2.0.0](#)
[lxqt-sudo-2.0.0](#)
[lxqt.openssh-askpass-2.0.1](#)
[lxqt-session-2.0.0](#)
[xdg-desktop-portal-lxqt-1.0.2](#)
[obconf-qt-0.16.4](#)
[LXQt Desktop Final Instructions](#)

38. LXQt Applications

[lximage-qt-2.0.1](#)
[lxqt-archiver-1.0.0](#)
[lxqt-notificationd-2.0.1](#)
[pavucontrol-qt-2.0.0](#)
[qps-2.9.0](#)
[qtermwidget-2.0.1](#)
[qterminal-2.0.1](#)
[screengrab-2.8.0](#)

XI. X Software

39. Office Programs

[AbiWord-3.0.5](#)
[Gnumeric-1.12.57](#)
[LibreOffice-24.8.0](#)

40. Graphical Web Browsers

[Epiphany-46.3](#)
[falkon-24.08.0](#)
[Firefox-128.1.0esr](#)
[Seamonkey-2.53.18.2](#)

[41. Other X-based Programs](#)

[Balsa-2.6.4](#)
[feh-3.10.3](#)
[FontForge-20230101](#)
[FreeRDP-3.7.0](#)
[Gimp-20240711](#)
[Gparted-1.6.0](#)
[Inkscape-1.3.2](#)
[rxvt-unicode-9.31](#)
[Thunderbird-128.1.0esr](#)
[Tigervnc-1.14.0](#)
[Transmission-4.0.6](#)
[xarchiver-0.5.4.23](#)
[xdg-utils-1.2.1](#)
[XScreenSaver-6.09](#)

[XII. Multimedia](#)

[42. Multimedia Libraries and Drivers](#)

[ALSA](#)
[alsa-lib-1.2.12](#)
[alsa-plugins-1.2.12](#)
[alsa-utils-1.2.12](#)
[alsa-tools-1.2.11](#)
[alsa-firmware-1.2.4](#)
[AudioFile-0.3.6](#)
[FAAC-1_30](#)
[FAAD2-2.11.1](#)
[fdk-aac-2.0.3](#)
[FLAC-1.4.3](#)
[frei0r-plugins-1.8.0](#)
[gavl-1.4.0](#)
[gstreamer-1.24.7](#)
[gst-plugins-base-1.24.7](#)
[gst-plugins-good-1.24.7](#)
[gst-plugins-bad-1.24.7](#)
[gst-plugins-ugly-1.24.7](#)
[gst-libav-1.24.7](#)
[id3lib-3.8.3](#)
[intel-media-24.2.5](#)
[intel-vaaapi-driver-2.4.1](#)
[Liba52-0.8.0](#)
[Libao-1.2.0](#)
[libaom-3.9.1](#)
[libass-0.17.3](#)
[libcanberra-0.30](#)
[libcddb-1.3.2](#)
[libcdio-2.1.0](#)
[libdvdcss-1.4.3](#)
[Libdvdrread-6.1.3](#)
[Libdvdnav-6.1.1](#)
[Libdv-1.0.0](#)
[libmad-0.15.1b](#)
[libmpeg2-0.5.1](#)
[libmusicbrainz-5.1.0](#)
[libogg-1.3.5](#)
[libplacebo-7.349.0](#)
[libsamplerate-0.2.2](#)
[libsndfile-1.2.2](#)
[libva-2.22.0](#)
[libvdpau-1.5](#)
[libvdpau-va-gl-0.4.2](#)
[libvorbis-1.3.7](#)
[libvpx-1.14.1](#)
[mlt-7.26.0](#)
[Opus-1.5.2](#)
[Pipewire-1.2.3](#)
[PulseAudio-17.0](#)
[SBC-2.0](#)
[sdl12-compat-1.2.68](#)
[SDL2-2.30.6](#)
[sound-theme-freedesktop-0.8](#)

[SoundTouch-2.3.3](#)
[Speex-1.2.1](#)
[Taglib-2.0.1](#)
[v4l-utils-1.28.1](#)
[Wireplumber-0.5.5](#)
[x264-20240812](#)
[x265-3.6](#)
[xine-lib-1.2.13](#)
[XviD-1.3.7](#)

43. Audio Utilities

[Audacious-4.4](#)
[CDParanoia-III-10.2](#)
[LAME-3.100](#)
[mpg123-1.32.7](#)
[pavucontrol-6.1](#)
[pnmixer-0.7.2](#)
[vorbis-tools-1.4.2](#)

44. Video Utilities

[FFmpeg-7.0.2](#)
[mpv-0.38.0](#)
[VLC-3.0.21](#)
[xine-ui-0.99.14](#)

45. CD/DVD-Writing Utilities

[Cdrdao-1.2.4](#)
[Cdrtools-3.02a09](#)
[dvd+rw-tools-7.1](#)
[libburn-1.5.6](#)
[libisoburn-1.5.6](#)
[libisofs-1.5.6](#)

XIII. Printing, Scanning and Typesetting

46. Printing

[Cups-2.4.10](#)
[cups-browsed-2.0.1](#)
[cups-filters-2.0.1](#)
[ghostscript-10.03.1](#)
[Gutenprint-5.3.4](#)
[libcupsfilters-2.0.0](#)
[libppd-2.0.0](#)

47. Scanning

[SANE-1.2.1](#)
[Simple-scan-46.0](#)

48. Standard Generalized Markup Language (SGML)

[sgml-common-0.6.3](#)
[docbook-3.1-dtd](#)
[docbook-4.5-dtd](#)
[OpenSP-1.5.2](#)
[OpenJade-1.3.2](#)
[docbook-dsssl-1.79](#)
[DocBook-utils-0.6.14](#)

49. Extensible Markup Language (XML)

[docbook-xml-4.5](#)
[docbook-xml-5.0](#)
[docbook-xml-5.1](#)
[docbook-xsl-nons-1.79.2](#)
[itstool-2.0.7](#)
[xmldt-0.0.29](#)

50. PostScript

[Enscript-1.6.6](#)
[ePDFView-gtk3-20200814](#)
[fop-2.9](#)
[MuPDF-1.24.8](#)
[paps-0.8.0](#)

51. Typesetting

[Setting the PATH for TeX Live](#)
[install-tl-ux](#)
[texlive-20240312-source](#)
[asymptote-2.91](#)
[biber-2.20](#)

[**A. Creative Commons License**](#)

[**B. The MIT License**](#)

[**Glossary**](#)

[**Index**](#)

Preface

Having helped out with Linux From Scratch for a short time, I noticed that we were getting many queries as to how to do things beyond the base LFS system. At the time, the only assistance specifically offered relating to LFS were the LFS hints (<https://www.linuxfromscratch.org/hints>). Most of the LFS hints are extremely good and well written but I (and others) could still see a need for more comprehensive help to go Beyond LFS — hence BLFS.

BLFS aims to be more than the LFS-hints converted to XML although much of our work is based around the hints and indeed some authors write both hints and the relevant BLFS sections. We hope that we can provide you with enough information to not only manage to build your system up to what you want, whether it be a web server or a multimedia desktop system, but also that you will learn a lot about system configuration as you go.

Thanks as ever go to everyone in the LFS/BLFS community; especially those who have contributed instructions, written text, answered questions and generally shouted when things were wrong!

Finally, we encourage you to become involved in the community; ask questions on the mailing list or news gateway and join in the fun on #lfs and #lfs-support at Libera. You can find more details about all of these in the [Introduction](#) section of the book.

Enjoy using BLFS.

Mark Hymers
markh <at> linuxfromscratch.org
BLFS Editor (July 2001–March 2003)

I still remember how I found the BLFS project and started using the instructions that were completed at the time. I could not believe how wonderful it was to get an application up and running very quickly, with explanations as to why things were done a certain way. Unfortunately, for me, it wasn't long before I was opening applications that had nothing more than "To be done" on the page. I did what most would do, I waited for someone else to do it. It wasn't too long before I am looking through Bugzilla for something easy to do. As with any learning experience, the definition of what was easy kept changing.

We still encourage you to become involved as BLFS is never really finished. Contributing or just using, we hope you enjoy your BLFS experience.

Larry Lawrence
larry <at> linuxfromscratch.org
BLFS Editor (March 2003–June 2004)

The BLFS project is a natural progression of LFS. Together, these projects provide a unique resource for the Open Source Community. They take the mystery out of the process of building a complete, functional software system from the source code contributed by many talented individuals throughout the world. They truly allow users to implement the slogan "*Your distro, your rules.*"

Our goal is to continue to provide the best resource available that shows you how to integrate many significant Open Source applications. Since these applications are constantly updated and new applications are developed, this book will never be complete. Additionally, there is always room for improvement in explaining the nuances of how to install the different packages. To make these improvements, we need your feedback. I encourage you to participate on the different mailing lists, news groups, and IRC channels to help meet these goals.

Bruce Dubbs
bdubbs <at> linuxfromscratch.org
BLFS Editor (June 2004–December 2006 and February 2011–now)

My introduction to the [B]LFS project was actually by accident. I was trying to build a GNOME environment using some how-tos and other information I found on the web. A couple of times I ran into some build issues and Googling pulled up some old BLFS mailing list messages. Out of curiosity, I visited the Linux From Scratch web site and shortly thereafter was hooked. I've not used any other Linux distribution for personal use since.

I can't promise anyone will feel the sense of satisfaction I felt after building my first few systems using [B]LFS instructions, but I sincerely hope that your BLFS experience is as rewarding for you as it has been for me.

The BLFS project has grown significantly the last couple of years. There are more package instructions and related dependencies than ever before. The project requires your input for continued success. If you discover that you enjoy building BLFS, please consider helping out in any way you can. BLFS requires hundreds of hours of maintenance to keep it even semi-current. If you feel confident enough in your editing skills, please consider joining the BLFS team. Simply contributing to the mailing list discussions with sound advice and/or providing patches to the book's XML will probably result in you receiving an invitation to join the team.

Foreword

This version of the book is intended to be used when building on top of a system built using the LFS book. Every effort has been made to ensure accuracy and reliability of the instructions. Many people find that using the instructions in this book after building the current stable or development version of LFS provides a stable and very modern Linux system.

Enjoy!

Randy McMurchy
August 24th, 2008

Who Would Want to Read this Book

This book is mainly aimed at those who have built a system based on the LFS book. It will also be useful for those who are using other distributions, and for one reason or another want to manually build software and need some assistance. Note that the material in this book, in particular the dependency listings, assumes that you are using a basic LFS system with every package listed in the LFS book already installed and configured. BLFS can be used to create a range of diverse systems and so the target audience is probably as wide as that of the LFS book. If you found LFS useful, you should also like this!

Since Release 7.4, the BLFS book version has matched the LFS book version. This book may be incompatible with a previous or later release of the LFS book.

Organization

This book is divided into the following fourteen parts.

Part I - Introduction

This part contains essential information which is needed to understand the rest of the book.

Part II - Post LFS Configuration and Extra Software

Here we introduce basic configuration and security issues. We also discuss a range of text editors, file systems, and shells which aren't covered in the main LFS book.

Part III - General Libraries and Utilities

In this section we cover libraries which are often needed throughout the book, as well as system utilities. Information on programming (including recompiling GCC to support its full range of languages) concludes this part.

Part IV - Basic Networking

Here we explain how to connect to a network when you aren't using the simple static IP setup presented in the main LFS book. Networking libraries and command-line networking tools are also covered here.

Part V - Servers

Here we show you how to set up mail and other servers (such as FTP, Apache, etc.).

Part VI - X + Window Managers

This part explains how to set up a basic X Window System, along with some generic X libraries and Window managers.

Part VII - KDE

This part is for those who want to use the K Desktop Environment, or parts of it.

Part VIII - GNOME

GNOME is the main alternative to KDE in the Desktop Environment arena.

Part IX - Xfce

Xfce is a lightweight alternative to GNOME and KDE.

Part X - LXQt

LXDE is another lightweight alternative to GNOME and KDE.

Part XI - More X Software

Office programs and graphical web browsers are important to most people. They, and some generic X software, can be found in this part of the book.

Part XII - Multimedia

Here we cover multimedia libraries and drivers, along with some audio, video, and CD-writing programs.

Part XIII - Printing, Scanning and Typesetting (PST)

This part covers document handling, from applications like Ghostscript, CUPS, and DocBook, all the way to texlive.

Appendices

The Appendices present information which doesn't belong in the body of book; they are included as reference material. The glossary of acronyms is a handy feature.

Part I. Introduction

Chapter 1. Welcome to BLFS

The Beyond Linux From Scratch book is designed to carry on from where the LFS book leaves off. But unlike the LFS book, it isn't designed to be followed straight through. Reading the [Which sections of the book?](#) part of this chapter should help guide you through the book.

Please read most of this part of the book carefully as it explains quite a few of the conventions used throughout the book.

Which Sections of the Book Do I Want?

Unlike the Linux From Scratch book, BLFS isn't designed to be followed in a linear manner. LFS provides instructions on how to create a base system which can become anything from a web server to a multimedia desktop system. BLFS attempts to guide you in the process of going from the base system to your intended destination. Choice is very much involved.

Everyone who reads this book will want to read certain sections. The [Introduction](#), which you are currently reading, contains generic information. Take special note of the information in [Chapter 2, Important Information](#), as this contains comments about how to unpack software, issues related to the use of different locales, and various other considerations which apply throughout the book.

The part on [Post LFS Configuration and Extra Software](#) is where most people will want to turn next. This deals not only with configuration, but also with Security ([Chapter 4, Security](#)), File Systems ([Chapter 5, File Systems and Disk Management](#) -- including GRUB for UEFI), Text Editors ([Chapter 6, Text Editors](#)), and Shells ([Chapter 7, Shells](#)). Indeed, you may wish to reference some parts of this chapter (especially the sections on Text Editors and File Systems) while building your LFS system.

Following these basic items, most people will want to at least browse through the [General Libraries and Utilities](#) part of the book. This contains information on many items which are prerequisites for other sections of the book, as well as some items (such as [Chapter 13, Programming](#)) which are useful in their own right. You don't have to install all of the libraries and packages found in this part; each BLFS installation procedure tells you which other packages this one depends upon. You can choose the program you want to install, and see what it needs. (Don't forget to check for nested dependencies!)

Likewise, most people will probably want to look at the [Networking](#) section. It deals with connecting to the Internet or your LAN ([Chapter 14, Connecting to a Network](#)) using a variety of methods such as DHCP and PPP, and with items such as Networking Libraries ([Chapter 17, Networking Libraries](#)), plus various basic networking programs and utilities.

Once you have dealt with these basics, you may wish to configure more advanced network services. These are dealt with in the [Servers](#) part of the book. Those wanting to build servers should find a good starting point there. Note that this section also contains information on several database packages.

The next twelve chapters deal with desktop systems. This portion of the book starts with a part talking about [Graphical Components](#). This part also deals with some generic X-based libraries ([Chapter 25, Graphical Environment Libraries](#)). After that, [KDE](#), [GNOME](#), [Xfce](#), and [LXQt](#) are given their own parts, followed by one on [X Software](#).

The book then moves on to deal with [Multimedia](#) packages. Note that many people may want to use the [ALSA](#) instructions from this chapter when first starting their BLFS journey; the instructions are placed here because it is the most logical place for them.

The final part of the main BLFS book deals with [Printing, Scanning and Typesetting](#). This is useful for most people with desktop systems, but even those who are creating dedicated server systems may find it useful.

We hope you enjoy using BLFS. May you realize your dream of building the perfectly personalized Linux system!

Conventions Used in this Book

Typographical Conventions

To make things easy to follow, a number of conventions are used throughout the book. Here are some examples:

```
./configure --prefix=/usr
```

This form of text should be typed exactly as shown unless otherwise noted in the surrounding text. It is also used to identify references to specific commands.

```
install-info: unknown option  
`--dir-file=/mnt/lfs/usr/info/dir'
```

This form of text (fixed width font) shows screen output, probably the result of issuing a command. It is also used to show filenames such as `/boot/grub/grub.conf`

Note

Please configure your browser to display fixed-width text with a good monospaced font, with which you can distinguish the glyphs of `111` or `000` clearly.

Emphasis

This form of text is used for several purposes, but mainly to emphasize important points, or to give examples of what to type.

<https://www.linuxfromscratch.org/>

This form of text is used for hypertext links external to the book, such as HowTos, download locations, websites, etc.

[seamonkey-2.53.18.2](#)

This form of text is used for links internal to the book, such as another section describing a different package.

```
cat > $LFS/etc/group << "EOF"  
root:x:0:  
bin:x:1:  
.....  
EOF
```

This style is mainly used when creating configuration files. The first command (in bold) tells the system to create the file `$LFS/etc/group` from whatever is typed on the following lines, until the sequence EOF is encountered. Therefore, this whole section is usually typed exactly as shown. Remember, copy and paste is your friend!

<REPLACED TEXT>

This form of text is used to encapsulate text that should be modified, and is not to be typed as shown, or copied and pasted. The angle brackets are not part of the literal text; they are part of the substitution.

root

This form of text is used to show a specific system user or group reference in the instructions.

Conventions Used for Package Dependencies

When new packages are created, the software's authors depend on prior work. In order to build a package in BLFS, these dependencies must be built before the desired package can be compiled. For each package, prerequisites are listed in one or more separate sections: Required, Recommended, and Optional.

Required Dependencies

These dependencies are the bare minimum needed to build the package. Packages in LFS, and the required dependencies of these required packages, are omitted from this list. Always remember to check for nested dependencies. If a dependency is said to be “runtime,” it is not needed for building the package, but only to use it after installation.

Recommended Dependencies

These are dependencies the BLFS editors have determined are important to give the package reasonable capabilities. If a recommended dependency is not said to be "runtime," package installation instructions assume it is installed. If it is not installed, the instructions may require modification, to accommodate the missing package. A recommended "runtime" dependency does not need to be installed before building the package, but must be built afterwards for running the package with reasonable capabilities.

Optional Dependencies

These are dependencies the package *may* use. Integration of optional dependencies may be automatic by the package, or additional steps not presented by BLFS may be necessary. Optional dependencies are sometimes listed without explicit BLFS instructions. In this case you must determine how to perform the installation yourself.

Conventions Used for Kernel Configuration Options

Some packages require specific kernel configuration options. The general layout for these looks like this:

Master section --->	
Subsection --->	
[*] Required parameter	[REQU_PAR]
<*> Required parameter (not as module)	[REQU_PAR_NMOD]
<*/M> Required parameter (could be a module)	[REQU_PAR_MOD]
<M> Required parameter (as a module)	[REQU_PAR_MOD_ONLY]
< /*/M> Optional parameter	[OPT_PAR]
< /M> Optional parameter (as a module if enabled)	[OPT_PAR_MOD_ONLY]
[] Incompatible parameter	[INCOMP_PAR]
< > Incompatible parameter (even as module)	[INCOMP_PAR_MOD]

[...] on the right gives the symbolic name of the option, so you can easily check whether it is set in your `.config` file. Note that the `.config` file contains a `CONFIG_` prefix before all symbolic names. The meaning of the various entries is:

Master section	top level menu item
Subsection	submenu item
Required parameter	the option can either be built-in, or not selected: it must be selected
Required parameter (not as module)	the option can be built-in, a module, or not selected (tri-state): it must be selected as built-in
Required parameter (could be a module)	the option can be built-in, a module, or not selected: it must be selected, either as built-in or as a module
Required parameter (as a module)	the option can be built-in, a module, or not selected: it must be selected as a module; selecting it as built-in may cause unwanted effects
Optional parameter	the option can be built-in, a module, or not selected: it may be selected as a module or built-in if you need it for driving the hardware or optional kernel features
Optional parameter (as a module if enabled)	the option can be built-in, a module, or not selected: it may be selected as a module if you need it for driving the hardware or optional kernel features, but selecting it as built-in may cause unwanted effects
Incompatible parameter	the option can either be built-in or not selected: it must <i>not</i> be selected
Incompatible parameter (even as module)	the option can be built-in, a module, or not selected: it must <i>not</i> be selected

Note that, depending on other selections, the angle brackets (`<>`) in the configuration menu may appear as braces (`{}`) if the option cannot be unselected, or even as dashes (`-*-` or `-M-`), when the choice is imposed. The help text describing the option specifies the other selections on which this option relies, and how those other selections are set.

The letter in blue is the hotkey for this option. If you are running `make menuconfig`, you can press a key to quickly traverse all the options with this key as the hotkey on the screen.

SBU values in BLFS

As in LFS, each package in BLFS has a build time listed in Standard Build Units (SBUs). These times are relative to the time it took to build binutils in LFS, and are intended to provide some insight into how long it will take to build a package. Most times listed are for a single processor or core to build the package. In some cases, large, long running builds tested on multi-core systems have SBU times listed with comments such as '(parallelism=4)'. These values indicate testing was done using multiple cores. Note that while this speeds up the build on systems with the appropriate hardware, the speedup is not linear and to some extent depends on the individual package and the specific hardware used.

For packages which use ninja (i.e., anything using meson) or rust, by default all cores are used; similar comments will be seen on such packages even when the build time is minimal.

Where even a parallel build takes more than 15 SBU, on certain machines the time may be considerably greater even when the build does not use swap. In particular, different micro-architectures will build some files at different relative speeds, and this can introduce delays when certain make targets wait for another file to be created. Where a large build uses a lot of

C++ files, processors with Simultaneous Multi Threading will share the Floating Point Unit and can take 45% longer than when using four 'prime' cores (measured on an intel i7 using taskset and keeping the other cores idle).

Some packages do not support parallel builds; for these, the make command must specify -j1. Packages that are known to impose such limits are so marked in the text.

Book Version

This is BLFS-BOOK version 12.2 dated September 1st, 2024. This is the 12.2-systemd branch of the BLFS book, currently targeting the LFS 12.2-systemd book. For development versions, if this version is older than a month, it's likely that your mirror hasn't been synchronized recently and a newer version is probably available for download or viewing. Check one of the mirror sites at <https://www.linuxfromscratch.org/mirrors.html> for an updated version.

Mirror Sites

The BLFS project has a number of mirrors set up world-wide to make it easier and more convenient for you to access the website. Please visit the <https://www.linuxfromscratch.org/mirrors.html> website for the list of current mirrors.

Getting the Source Packages

Within the BLFS instructions, each package has two references for finding the source files for the package—an HTTP link and an FTP link (some packages may only list one of these links). Every effort has been made to ensure that these links are accurate. However, the World Wide Web is in continuous flux. Packages are sometimes moved or updated and the exact URL specified is not always available.

To overcome this problem, the BLFS Team, with the assistance of Oregon State University Open Source Lab, has made an HTTP/FTP site available through world wide mirrors. See <https://www.linuxfromscratch.org/blfs/download.html#sources> for a list. These sites have all the sources of the exact versions of the packages used in BLFS. If you can't find the BLFS package you need at the listed addresses, get it from these sites.

We would like to ask a favor, however. Although this is a public resource for you to use, please do not abuse it. We have already had one unthinking individual download over 3 GB of data, including multiple copies of the same files that are placed at different locations (via symlinks) to make finding the right package easier. This person clearly did not know what files he needed and downloaded everything. The best place to download files is the site or sites set up by the source code developer. Please try there first.

Change Log

Current release: 12.2 – September 1st, 2024

Changelog Entries:

- September 1st, 2024
 - [bdubbs] - Release of BLFS-12.2.
- August 30th, 2024
 - [renodr] - Update to libreoffice-24.8.0.3. Fixes [#20263](#).
- August 27th, 2024
 - [renodr] - Fix building Brasero with GCC-14. Fixes [#20278](#).
 - [renodr] - Restore libgedit-gtksourceview as it's needed by gedit.
 - [renodr] - Update to pipewire-1.2.3. Fixes [#20264](#).
 - [bdubbs] - Update to kde-gear-24.08.0. Includes falkon, kate, and k3b. Fixes [19954](#).
 - [renodr] - Update to gnome-desktop-44.1. Fixes [#20255](#).
- August 26th, 2024
 - [bdubbs] - Archive libquicktime and transcode.
 - [bdubbs] - Update to mlt-7.26.0. Fixes [20272](#).
- August 25th, 2024
 - [bdubbs] - Update to mc-4.8.32. Fixes [20267](#).
 - [bdubbs] - Update to Vulkan-Headers 1.3.294 and Vulkan-Loader-1.3.294. Fixes [20269](#).
- August 23th, 2024

- [bdubbs] - Update to libbytesize-2.11. Fixes [#20252](#).
 - [bdubbs] - Update to LVM2-2.03.26. Fixes [#20266](#).
- August 23rd, 2024
 - [bdubbs] - Update to gstreamer, gst-plugins-base, gst-plugins-good, gst-plugins-bad, gst-plugins-ugly, and gst-libav versions 1.24.7. Fixes [#20256](#).
 - [renodr] - Fix building ghostscript with GCC-14. Fixes [#20265](#).
- August 21st, 2024
 - [renodr] - Update to numpy-2.1.0. Fixes [#20247](#).
 - [renodr] - Fix building Subversion with GCC 14. Fixes [#20257](#).
 - [bdubbs] - Update to Net-DNS-1.46 (Perl module). Fixes [#20253](#).
- August 20th, 2024
 - [renodr] - Fix three security vulnerabilities in p7zip. Fixes [#20251](#).
- August 19th, 2024
 - [xry111] - Disable buggy expat support in libarchive-3.7.4. Fixes [#20249](#).
 - [xry111] - Patch libxml2-2.13.3 to fix an issue causing bogus warning with `xmlcatalog --create`. Fixes [#20248](#).
- August 18th, 2024

 - [renodr] - Update to dtc-1.7.1. Fixes [#20245](#).
 - [renodr] - Update to pax-20240817. Fixes [#20244](#).
 - [renodr] - Update to vulkan-headers and vulkan-loader 1.3.293. Fixes [#20242](#).
 - [renodr] - Update to gtk-4.14.5. Fixes [#20243](#).
 - [renodr] - Fix a segmentation fault in Libreoffice caused by the Boost patch. Fixes [#20229](#).
- August 17th, 2024
 - [bdubbs] - Update to librsvg-2.58.3. Fixes [#20240](#).
 - [xry111] - Update to rust-bindgen-0.70.0. Fixes [#20241](#).
- August 16th, 2024
 - [renodr] - Fix a regression in WebKitGTK that causes crashes when rendering some websites. Fixes [#20235](#).
 - [renodr] - Update to asymptote-2.91. Fixes [#20239](#).
 - [renodr] - Update cups-browsed and cups-filters to 2.0.1. Fixes [#20233](#).
 - [renodr] - Update to dovecot-2.3.17.1 (Security Update). Fixes [#20231](#).
 - [renodr] - Update to enchant-2.8.2. Fixes [#20230](#).
 - [renodr] - Fix building Libreoffice with boost-1.86.0.
 - [renodr] - Update to boost-1.86.0. Fixes [#20229](#).
 - [bdubbs] - Update to unbound-1.21.0. Fixes [#20238](#).
 - [bdubbs] - Update to btrfs-progs-6.10.1. Fixes [#20237](#).
 - [bdubbs] - Update to gnutls-3.8.7.1. Fixes [#20236](#).
- August 15th, 2024
 - [renodr] - Update to sentry_sdk-2.13.0 (Python Module). Fixes [#20062](#).
 - [renodr] - Update to x265-3.6. Fixes [#20234](#).
 - [bdubbs] - Update to xfce4-notifyd-0.9.6. Fixes [#20227](#).
 - [bdubbs] - Update to sqlite-autoconf-3460100. Fixes [#20226](#).
 - [bdubbs] - Update to freetype-2.13.3. Fixes [#20223](#).
 - [bdubbs] - Update to python3-3.12.5. Fixes [#20202](#).
- August 12th, 2024
 - [renodr] - Update to WebKitGTK-2.44.3. Fixes [#20225](#).
 - [thomas] - Update to xfburn-0.7.2. Fixes [#20224](#).
- August 12th, 2024
 - [renodr] - Update to x265-20240812. Fixes [#20221](#).
 - [renodr] - Update to x264-20240812. Fixes [#20221](#).
 - [renodr] - Update to ImageMagick-7.1.1-36. Fixes [#20222](#).
 - [renodr] - Update python module dependencies for BLFS 12.2. This includes alabaster-1.0.0, attrs-24.2.0, babel-2.16.0, certifi-2024.7.4, chardet-5.2.0, charset-normalizer-3.3.2, idna-3.7, markdown-3.6, msgpack-1.0.8,

sphinxcontrib-devhelp-2.0.0, sphinxcontrib-qthelp-2.0.0, sphinxcontrib-serializinghtml-2.0.0, and urllib3-2.2.2. Note that urllib3 and idna are security updates. Fixes [#20220](#).

- [renodr] - Update perl module dependencies for BLFS 12.2. This includes Alien-Build-2.83, Business-ISBN-Data-20240807.001, Date-Time-Locale-1.43, HTTP-Message-6.46, Path-Tiny-0.146, Term-Table-0.022, Test-Without-Module-0.23, Test2-Plugin-NoWarnings-0.10, and Text-CSV_XS-1.56. Fixes [#20219](#).
 - [renodr] - Replace Test2-Suite with Test-Simple. Fixes [#20219](#).
 - [renodr] - Update to Ixml-5.3.0 (Python Module). Fixes [#20215](#).
 - [renodr] - Update to NetworkManager-1.48.8. Fixes [#20212](#).
 - [renodr] - Update to sentry_sdk-2.12.0 (Python Module). Fixes [#20062](#).
 - [bdubbs] - Update to vim-9.1.0660 (sync with LFS). Addresses [#12241](#).
 - [bdubbs] - Update to highlight-4.13. Fixes [#20218](#).
 - [bdubbs] - Update to graphviz-12.1.0. Fixes [#20217](#).
 - [xry111] - Update to cbindgen-0.27.0. Fixes [#20214](#).
 - [thomas] - Update to vte-0.76.4. Fixes [#20213](#).
- August 11th, 2024
 - [bdubbs] - Update to kwayland, libkscreen, and layer-shell-qt 6.1.4 (for lxqt). Fixes [#19973](#).
 - [bdubbs] - Update to kconfig, kidletime, kwindowsystem, and solid 6.5.0 (for lxqt). Fixes [#19922](#).
 - [bdubbs] - Update to plasma-6.1.4. Fixes [#19973](#).
 - [bdubbs] - Add new package pulseaudio-qt-1.5.0 in support of the plasma-pa-6.1.4 package in the plasma-6.1.4 suite.
 - [rahul] - Update to protobuf-27.3. Fixes [#20163](#).
 - [rahul] - Update to pipewire-1.2.2. Fixes [#20165](#).
 - [bdubbs] - Update to kirigami-addons-1.4.0. Fixes [#20131](#).
 - [bdubbs] - Update to kf6-6.5.0. Includes extra-cmake-modules, and breeze-icons. Fixes [#19921](#).
 - August 9th, 2024
 - [renodr] - Update to FreeRDP-3.7.0. Fixes [#20211](#).
 - [renodr] - Update to postgresql-16.4 (Security Update). Fixes [#20210](#).
 - [renodr] - Update to Regexp-Common-2024080801 (Perl Module). Fixes [#20208](#).
 - [xry111] - Update to polkit-125. Fixes [#20207](#).
 - [xry111] - Update to rustc-1.80.1. Fixes [#20209](#).
 - August 8th, 2024
 - [bdubbs] - Update to mpg123-1.32.7. Fixes [#20205](#).
 - [bdubbs] - Update to xwayland-24.1.2. Fixes [#20203](#).
 - [bdubbs] - Update to libei-1.3.0. Fixes [#20204](#).
 - [bdubbs] - Update to doxygen-1.12.0. Fixes [#20201](#).
 - August 7th, 2024
 - [renodr] - Update to gnome-online-accounts-3.50.4. Fixes [#20200](#).
 - [renodr] - Update evolution and evolution-data-server to 3.52.4. Fixes [#20199](#).
 - [renodr] - Update to samba-4.20.4. Fixes [#20196](#).
 - [renodr] - Update to gnome-control-center-46.4. Fixes [#20193](#).
 - [renodr] - Update to pavucontrol-6.1. Fixes [#20185](#).
 - [renodr] - Update to thunderbird-128.1.0esr (Security Update). Fixes [#20153](#).
 - [renodr] - Update to firefox-128.1.0esr (Security Update). Fixes [#20194](#).
 - [renodr] - Update to spidermonkey-115.14.0 (Security Update). Fixes [#20195](#).
 - August 6th, 2024
 - [bdubbs] - Update to pyyaml-6.0.2. Fixes [#20198](#).
 - [bdubbs] - Update to hwdatas-0.385. Fixes [#20197](#).
 - [rahul] - Update to nss-3.103. Fixes [#20167](#).
 - [bdubbs] - Update to cython-3.0.11. Fixes [#20192](#).
 - August 5th, 2024
 - [renodr] - Update to gnome-user-docs-46.4. Fixes [#20191](#).
 - [renodr] - Update to mutter-46.4. Fixes [#20189](#).
 - [renodr] - Update to gnome-shell-46.4. Fixes [#20188](#).

- [renodr] - Update to ffmpeg-7.0.2. Fixes [#20184](#).
 - [renodr] - Update to epiphany-46.3. Fixes [#20183](#).
 - [renodr] - Update to gnome-bluetooth-46.1. Fixes [#20182](#).
 - [renodr] - Update to cmake-3.30.2. Fixes [#20180](#).
- August 4th, 2024
 - [bdubbs] - Update to sysmon-qt-2.0.1.
 - [bdubbs] - Update to cracklib-2.10.2. Fixes [#20187](#).
- August 3rd, 2024
 - [bdubbs] - Update to SDL2-2.30.6. Fixes [#20186](#).
 - [bdubbs] - Update to libadwaita-1.5.3. Fixes [#20181](#).
 - [bdubbs] - Update to libnvme-1.10. Fixes [#20179](#).
 - [bdubbs] - Update to Vulkan-Headers 1.3.292 and Vulkan-Loader-1.3.292. Fixes [#20114](#).
- August 2nd, 2024
 - [renodr] - Update to c-ares-1.33.0. Fixes [#20178](#).
 - [renodr] - Update to samba-4.20.3. Fixes [#20177](#).
 - [renodr] - Update to lcms2-2.16. Fixes [#20175](#).
 - [renodr] - Update to eog-45.4. Fixes [#20173](#).
 - [renodr] - Update to libshumate-1.2.3. Fixes [#20173](#).
 - [bdubbs] - Update to libFS-1.0.10, libXfont2-2.0.7, and libXtst-1.2.5 (Xorg libraries). Fixes [#20172](#).
 - [bdubbs] - Update to poppler-24.08.0. Fixes [#20169](#).
 - [bdubbs] - Update to gcc-14.2.0. Fixes [#20166](#).
 - [renodr] - Update to php-8.3.10. Fixes [#20171](#).
 - [renodr] - Update to mercurial-6.8.1. Fixes [#20170](#).
 - [renodr] - Update to abseil-cpp-20240722.0. Fixes [#20168](#).
- August 1st, 2024
 - [bdubbs] - Update to btrfs-progs-v6.10. Fixes [#20155](#).
 - [bdubbs] - Update to sphinx-8.0.2. Also updated to sphinxcontrib-applehelp-2.0.0. Fixes [#20117](#).
 - [bdubbs] - Update to gstreamer, gst-plugins-base, gst-plugins-good, gst-plugins-bad, gst-plugins-ugly, and gst-libav versions 1.24.6. Fixes [#19984](#).
 - [thomas] - Update to ISC Kea-dhcpd-2.6.1. Fixes [#20164](#).
 - [renodr] - Update to SPIRV-LLVM-Translator-18.1.3. Fixes [#20162](#).
- July 31st, 2024
 - [renodr] - Update to mesa-24.1.5. Fixes [#20159](#).
 - [renodr] - Update to HTML-Parser-3.83 (Perl Module). Fixes [#20157](#).
 - [renodr] - Update to libavif-1.1.1. Fixes [#20156](#).
 - [renodr] - Update to cryptsetup-2.7.4. Fixes [#20154](#).
 - [renodr] - Update to curl-8.9.1 (Security Update). Fixes [#20160](#).
 - [thomas] - Update to thunar-4.18.11. Fixes [#20158](#).
 - [renodr] - Update to systemd-256.4 (sync with LFS). Fixes [#20084](#).
- July 30th, 2024
 - [bdubbs] - Update to git-2.46.0. Fixes [#20152](#).
- July 29th, 2024
 - [bdubbs] - Update to links-2.30. Fixes [#20150](#).
 - [bdubbs] - Update to libX11-1.8.10 (Xorg library). Fixes [#20151](#).
- July 27th, 2024
 - [bdubbs] - Update to rpcbind-1.2.7. Fixes [#20145](#).
 - [xry111] - Update to rustc-1.80.0. Fixes [#20145](#).
- July 26th, 2024
 - [bdubbs] - Update to cracklib-2.10.1. Fixes [#20143](#).
 - [renodr] - Update to pangomm-2.54.0. Fixes [#20147](#).
 - [renodr] - Update to NetworkManager-1.48.6. Fixes [#20146](#).

- [renodr] - Update to pytest-8.3.2 (Python Modules). Fixes [#20144](#).
- July 25th, 2024
 - [bdubbs] - Update to tigervnc-1.14.0. Fixes [#20140](#).
 - [bdubbs] - Update to mupdf-1.24.8. Fixes [#20142](#).
 - [bdubbs] - Update to nss-3.102.1. Fixes [#20135](#).
 - [thomas] - Update to c-ares-1.32.3. Fixes [#20136](#).
 - [renodr] - Update to dvisvgm-3.4. Fixes [#20141](#).
 - [renodr] - Update to libtirpc-1.3.5. Fixes [#20139](#).
 - [renodr] - Update to libxml2-2.13.3 (Security Update). Fixes [#20136](#).
 - [renodr] - Update to node.js-20.16.0. Fixes [#20137](#).
 - [renodr] - Update to OpenJDK-22.0.2 (Security Update). Fixes [#20101](#).
- July 24th, 2024
 - [bdubbs] - Update to v4l-utils-1.28.1. Fixes [#20133](#).
 - [renodr] - Update to curl-8.9.0 (Security Update). Fixes [#20134](#).
 - [renodr] - Update to BIND-9.20.0. Fixes [#20132](#).
- July 23rd, 2024
 - [thomas] - Update to xfs-progs-6.9.0. Fixes [#20127](#).
 - [renodr] - Update to BIND-9.18.28 (Security Update). Fixes [#20130](#).
 - [renodr] - Update to evince-46.3.1. Fixes [#20129](#).
 - [renodr] - Update to SPIRV-Headers and SPIRV-Tools 1.3.290. Fixes [#20128](#).
- July 22nd, 2024
 - [renodr] - Update to fetchmail-6.4.39. Fixes [#20119](#).
 - [renodr] - Fix CVE-2023-43361 in vorbis-tools (Security Update). Fixes [#20125](#).
 - [renodr] - Fix building compface with GCC 14. Fixes [#20126](#).
 - [renodr] - Adapt QtWebEngine to use system ffmpeg. Fixes [#20102](#).
 - [bdubbs] - Update to intel-media-24.2.5/intel-gmmlib-22.4.1. Fixes [#20124](#).
 - [bdubbs] - Update to numpy-2.0.1 (Python module). Fixes [#20123](#).
- July 21st, 2024
 - [bdubbs] - Update to libcdio-paranoia-10.2+2.0.2 (Part of libcdio). Fixes [#20121](#).
 - [bdubbs] - Update to ldns-1.8.4. Fixes [#20120](#).
 - [bdubbs] - Update to wpa_supplicant-2.11. Fixes [#20118](#).
 - [bdubbs] - Update to fmt-11.0.2. Fixes [#20115](#).
 - [bdubbs] - Update to pytest-8.3.1 (Python module). Fixes [#20116](#).
- July 20th, 2024
 - [bdubbs] - Update to mupdf-1.24.7. Fixes [#20113](#).
 - [bdubbs] - Update to v4l-utils-1.28.0. Fixes [#20112](#).
 - [bdubbs] - Update to libnl-3.10.0. Fixes [#20110](#).
 - [bdubbs] - Update to sphinx-7.4.6 (Python module). Fixes [#20109](#).
- July 18th, 2024
 - [bdubbs] - Update to qt-5.15.14 (components). Fixes [#19442](#).
 - [renodr] - Update to exiv2-0.28.3 (Security Update). Fixes [#20105](#).
 - [bdubbs] - Update to xapian-core-1.4.26. Fixes [#20108](#).
- July 18th, 2024
 - [bdubbs] - Update to httpd-2.4.62 (Security Update). Fixes [#20103](#).
 - [bdubbs] - Update to cmake-3.30.1. Fixes [#20106](#).
 - [bdubbs] - Update to mesa-24.1.4. Fixes [#20104](#).
 - [bdubbs] - Archive gtk2. Fixes [#18531](#). Also archive pygtk, mplayer, gtk-engines, hexchat, and pidgin.
 - [bdubbs] - Archive python2. Fixes [#11549](#).
 - [bdubbs] - Update to gimp-20240711 (gimp-3.0 snapshot). Fixes [#19886](#).
- July 17th, 2024
 - [bdubbs] - Update to qemu-9.0.2. Fixes [#20098](#).

- [bdubbs] - Update to sphinx-7.4.5 (Python module). Fixes [#20100](#).
 - [bdubbs] - Update to asciidoc-10.2.1 (Python module). Fixes [#20099](#).
 - [renodr] - Archive libgrss.
 - [bdubbs] - Archive vsftpd.
- July 16th, 2024
 - [renodr] - Update to gsettings-desktop-schemas-46.1. Fixes [#20097](#).
 - [renodr] - Update to python-dbusmock-0.32.1 (Python Module). Fixes [#20093](#).
 - [renodr] - Update to gnome-keyring-46.2. Fixes [#20085](#).
 - [renodr] - Update to gvfs-1.54.2. Fixes [#20075](#).
 - [renodr] - Update to gnome-online-accounts-3.50.3. Fixes [#20074](#).
 - [xry111] - Update to hatchling-1.25.0 (Python dependency). Addresses [#18562](#).
 - [xry111] - Update to trove_classifiers-2024.7.2 (Python dependency). Addresses [#18562](#).
 - [renodr] - Update to evolution-3.52.3. Fixes [#20073](#).
 - [renodr] - Update to evolution-data-server-3.52.3. Fixes [#20073](#).
 - [renodr] - Update to gnome-control-center-46.3. Fixes [#20036](#).
 - [renodr] - Update to epiphany-46.2. Fixes [#20024](#).
 - [renodr] - Update to mutter-46.3.1. Fixes [#20023](#).
 - [renodr] - Update to gnome-shell-46.3.1. Fixes [#20022](#).
 - [renodr] - Update to gexiv2-0.14.3. Fixes [#20017](#).
 - [renodr] - Update to mercurial-6.8. Fixes [#20060](#).
 - [renodr] - Archive telepathy-glib because nothing uses it anymore.
 - [renodr] - Add gnome-connections to the book. Fixes [#19960](#).
 - [renodr] - Archive Vinagre. This is a part of [#19960](#).
 - [renodr] - Add FreeRDP to the book in support of gnome-connections. This is a part of [#19960](#).
 - [renodr] - Update to Thunderbird-128.0esr (Security Update). Fixes [#19717](#).
 - [xry111] - Archive typing_extensions (Python dependency).
 - [xry111] - Update to setuptools_scm-8.1.0 (Python dependency). Addresses [#18562](#).
 - [bdubbs] - Update to c-ares-1.32.2. Fixes [#20096](#).
 - [bdubbs] - Update to sphinx-7.4.4 (Python module). Fixes [#20095](#).
 - [bdubbs] - Update to babel-2.15.0 (Python module). Needed by sphinx-7.4.4.
- July 15th, 2024
 - [bdubbs] - Update to xfburn-0.7.1. Fixes [#20094](#).
 - [bdubbs] - Update to IO-Socket-SSL-2.088 (Perl Module). Fixes [#20092](#).
- July 14th, 2024
 - [bdubbs] - Add konversation to the kf6 applications in the book.
 - [rahul] - Update to nodejs-20.15.1. Fixes [#19983](#).
 - [rahul] - Update to bluez-5.77. Fixes [#20058](#).
 - [rahul] - Update to nss-3.102. Fixes [#20012](#).
 - [bdubbs] - Update to SDL2-2.30.5. Fixes [#20091](#).
 - [bdubbs] - Update to make-ca-1.14. Fixes [#20090](#).
 - [bdubbs] - Update to cracklib-2.10.0. Fixes [#20089](#).
 - [bdubbs] - Update to Vulkan-Headers and Vulkan-Loader 1.3.290. Fixes [#20087](#).
 - [bdubbs] - Update to imlib2-1.12.3. Fixes [#20088](#).
- July 13th, 2024
 - [bdubbs] - Update to libreoffice-24.2.5.2. Fixes [#20078](#).
 - [bdubbs] - Update to pipewire-1.2.1. Fixes [#20086](#).
 - [bdubbs] - Update to librsvg-2.58.2. Fixes [#20083](#).
 - [bdubbs] - Update to LVM2.2.03.25. Fixes [#20082](#).
 - [bdubbs] - Update to libavif-1.1.0. Fixes [#20080](#).
 - [bdubbs] - Update to gdb-15.1. Fixes [#20054](#).
- July 12th, 2024
 - [bdubbs] - Update to mupdf-1.24.6. Fixes [#20069](#).

- [bdubbs] - Update to xwayland-24.1.1. Fixes [#20067](#).
 - [bdubbs] - Update to wireshark-4.2.6. Fixes [#20065](#).
 - [bdubbs] - Update to c-ares-1.32.1. Fixes [#20064](#).
 - [bdubbs] - Update to unix-tree-2.1.3. Fixes [#20063](#).
 - [renodr] - Update to firefox-128.0esr (Security Update). Fixes [#20056](#).
 - [renodr] - Update to mozjs-115.13.0. Fixes [#20076](#).
 - [renodr] - Update to gtk+-3.24.43 (Security Update). Fixes [#20068](#).
 - [renodr] - Update to ruby-3.3.4. Fixes [#20061](#).
 - [renodr] - Update to NetworkManager-1.48.4. Fixes [#20052](#).
 - [renodr] - Fix building libgsf with libxml2-2.13. Fixes [#20071](#).
 - [renodr] - Update to asymptote-2.90. Fixes [#19989](#).
 - [renodr] - Update to libwacom-2.12.2. Fixes [#19987](#).
 - [renodr] - Update to systemd-256.1. Fixes [#19967](#).
 - [ken] - Patch gimp-2.10.38 to build with gcc14. Fixes [19886](#).
 - [thomas] - Upgrade to nano-8.1. Fixes [20081](#).
 - [thomas] - Upgrade to xterm-393. Fixes [20079](#).
 - [ken] - Patch gtk+-2.24.33 to build with gcc14.. Fixes [19887](#).
- July 11th, 2024
 - [timtas] - Update to exim-4.98. Fixes [#20066](#).
 - [bdubbs] - Update to IO-Socket-SSL-2.087 (Perl Module). Fixes [#20059](#).
 - [bdubbs] - Update to glib-2.80.4. Fixes [#20057](#).
 - [bdubbs] - Update to SCons-4.8.0. Fixes [#20055](#).
 - July 8th, 2024
 - [rahul] - Update to mesa-24.1.3. Fixes [#20041](#).
 - [rahul] - Update to php-8.3.9. Fixes [#20049](#).
 - [rahul] - Update to libxslt-1.1.42. Fixes [#20046](#).
 - [rahul] - Update to graphviz-12.0.0. Fixes [#20047](#).
 - [rahul] - Update to p11-kit-0.25.5. Fixes [#20044](#).
 - [rahul] - Update to libxml2-2.13.2. Fixes [#20045](#).
 - [rahul] - Update to gnutls-3.8.6. Fixes [#20038](#).
 - [rahul] - Update to c-ares-1.32.0. Fixes [#20050](#).
 - July 6th, 2024
 - [bdubbs] - Update to sentry_sdk-2.7.1. Fixes [#19910](#).
 - [bdubbs] - Update to fmt-11.0.1. Fixes [#20053](#).
 - July 5th, 2024
 - [bdubbs] - Update to hwdatalog-0.384. Fixes [#20048](#).
 - [bdubbs] - Update to IO-Socket-SSL-2.086 (Perl Module). Fixes [#20042](#).
 - [bdubbs] - Update to sysstat-12.7.6. Fixes [#20040](#).
 - [bdubbs] - Update to pinentry-1.3.1. Fixes [#20039](#).
 - July 4th, 2024
 - [timtas] - Update to httpd-2.4.61 (Security Update). Fixes [#20031](#).
 - July 3rd, 2024
 - [bdubbs] - Update to libass-0.17.3. Fixes [#20037](#).
 - [bdubbs] - Update to libva-2.22.0. Fixes [#20034](#).
 - [bdubbs] - Update to libplacebo-7.349.0 (note unusual version number change). Fixes [#20033](#).
 - [bdubbs] - Update to p11-kit-0.25.4. Fixes [#20032](#).
 - [bdubbs] - Update to unix-tree-2.1.2. Fixes [#20030](#).
 - [bdubbs] - Update to libqalculate-5.2.0. Fixes [#20029](#).
 - [bdubbs] - Update to fmt-11.0.0. Fixes [#20028](#).
 - July 1st, 2024
 - [bdubbs] - Update to poppler-24.07.0. Fixes [#19947](#).
 - [bdubbs] - Update to SPIRV-LLVM-Translator-18.1.2. Fixes [#20025](#).

- [bdubbs] - Update to openssh-9.8p1 (Security Update). Fixes [#20027](#).
 - [bdubbs] - Update to feh-3.10.3. Fixes [#20026](#).
 - [bdubbs] - Update to shadow-4.16.0. Fixes [#19965](#).
- June 29th, 2024
 - [bdubbs] - Update to highlight-4.12. Fixes [#20021](#).
 - [rahul] - Update to wireplumber-0.5.5. Fixes [#20020](#).
 - [bdubbs] - Update to libadwaita-1.5.2. Fixes [#20015](#).
 - [bdubbs] - Update to Vulkan-Headers and Vulkan-Loader 1.3.289. Fixes [#20014](#).
 - [bdubbs] - Update to libndp-1.9. Fixes [#20013](#).
- June 28th, 2024
 - [bdubbs] - Update to qtermwidget and qterminal-2.0.1. Fixes [#20007](#).
 - [bdubbs] - Update to libjxl-0.10.3. Fixes [#20010](#).
 - [bdubbs] - Update to libinput-1.26.1 (Xorg input driver). Fixes [#20009](#).
 - [bdubbs] - Update to mupdf-1.24.5. Fixes [#20008](#).
 - [bdubbs] - Update to harfbuzz-9.0.0. Fixes [#20011](#).
- June 27th, 2024
 - [rahul] - Update to wireplumber-0.5.4. Fixes [#20004](#).
 - [rahul] - Update to pipewire-1.2.0. Fixes [#20005](#).
 - [bdubbs] - Update to kirigami-addons-1.3.0. Fixes [#20002](#).
 - [bdubbs] - Update to btrfs-progs-v6.9.2. Fixes [#20001](#).
 - [bdubbs] - Update to xmlto-0.0.29. Fixes [#20006](#).
 - [bdubbs] - Update to krb5-1.21.3 (Security Update). Fixes [#20000](#).
 - [bdubbs] - Update to libdrm-2.4.122. Fixes [#20003](#).
 - [bdubbs] - Update to glslang-14.3.0. Fixes [#19999](#).
 - [bdubbs] - Update to lua-5.4.7. Fixes [#19998](#).
 - [bdubbs] - Update to protobuf-27.2. Fixes [#19997](#).
- June 25th, 2024
 - [bdubbs] - Update to qca-2.3.9. Fixes [#19996](#).
 - [bdubbs] - Update to NetworkManager-1.48.2. Fixes [#19995](#).
 - [bdubbs] - Update to libassuan-3.0.1. Fixes [#19994](#).
 - [thomas] - Update to btrfs-progs-6.9.1. Fixes [#19993](#).
- June 23rd, 2024
 - [bdubbs] - Update to guile-3.0.10. Fixes [#19992](#).
- June 22nd, 2024
 - [bdubbs] - Update to emacs-29.4 (Security Update). Fixes [#19991](#).
 - [bdubbs] - Update to mupdf-1.24.4. Fixes [#19990](#).
 - [bdubbs] - Update to pycairo-1.26.1 (Python module). Fixes [#19988](#).
 - [bdubbs] - Update to libksba-1.6.7. Fixes [#19986](#).
 - [timtas] - Update to samba-4.20.2. Fixes [#19979](#).
- June 21st, 2024
 - [bdubbs] - Update to libdisplay-info-0.2.0. Fixes [#19982](#).
- June 20th, 2024
 - [bdubbs] - Update to mesa-24.1.2. Fixes [#19980](#).
 - [bdubbs] - Update to libxslt-1.1.41. Fixes [#19978](#).
 - [bdubbs] - Update to libxml2-2.13.1. Fixes [#19977](#).
 - [bdubbs] - Update to libgcrypt-1.11.0. Fixes [#19975](#).
 - [bdubbs] - Update to libgpg-error-1.50. Fixes [#19976](#).
- June 19th, 2024
 - [xry111] - Remove qtlocation from qt5 and qt5-components. Addresses [#19442](#).
 - [bdubbs] - Update to uhttpmock-0.11.0. Fixes [#19972](#).
 - [bdubbs] - Update to psutil-6.0.0 (Python module). Fixes [#19971](#).

- [bdubbs] - Update to c-ares-1.31.0. Fixes [#19969](#).
 - [bdubbs] - Update to libassuan-3.0.0. Fixes [#19966](#).
 - [bdubbs] - Update to qt6-6.7.2 and qtwebengine-6.7.2. Fixes [#19970](#).
 - [timtas] - Update to cups-2.4.10. Fixes [#19974](#).
- June 18th, 2024
 - [rahul] - Update to cryptsetup-2.7.3 (Security Update). Fixes [#19964](#).
- June 17th, 2024
 - [timtas] - Update to thunderbird-115.12.0. Fixes [#19958](#).
 - [renodr] - Update to intel-media-driver-24.1.5. Fixes [#19704](#).
 - [bdubbs] - Update to numpy-2.0.0 (Python module). Fixes [#19962](#).
 - [bdubbs] - Update to icewm-3.6.0. Fixes [#19963](#).
 - [bdubbs] - Update to nettle-3.10. Fixes [#19961](#).
- June 15th, 2024
 - [bdubbs] - Update to libinput-1.26.0 (Xorg input driver). Fixes [#19905](#).
 - [bdubbs] - Update to SDL2-2.30.3. Fixes [#19904](#).
 - [bdubbs] - Update to protobuf-27.1. Fixes [#19902](#).
 - [bdubbs] - Update Vulkan-Headers and Vulkan-Loader to version 1.3.288. Fixes [#19957](#).
 - [bdubbs] - Update to enchant-2.8.1. Fixes [#19952](#).
 - [bdubbs] - Update to freeglut-3.6.0. Fixes [#19949](#).
 - [bdubbs] - Update to mercurial-6.7.4. Fixes [#19948](#).
 - [ken] - Patch OpenSP to compile with gcc-14. Fixes [#19956](#).
- June 14th, 2024
 - [xry111] - Update to systemd-256. Fixes [#19940](#).
 - [xry111] - Patch libxml2-2.13.0 to fix several issues breaking various downstream packages. Fixes [#19955](#).
 - [bdubbs] - Update to audacious/audacious-plugins-4.4. Fixes [#19950](#).
 - [bdubbs] - Update to libxslt-1.1.40. Fixes [#19946](#).
 - [bdubbs] - Update to tcsh-6.24.13. Fixes [#19945](#).
 - [bdubbs] - Update to Python3-3.12.4 (Security Update). Fixes [#19909](#).
 - [xry111] - Update to rustc-1.79.0. Fixes [#19953](#).
- June 12th, 2024
 - [bdubbs] - Update to alsa-lib alsa-plugins alsa-utils 1.2.12. Fixes [#19939](#).
 - [bdubbs] - Update to ruby-3.3.3. Fixes [#19941](#).
 - [bdubbs] - Update to XML-LibXSLT-2.003000 (Perl module). Fixes [#19942](#).
 - [renodr] - Update to libxml2-2.13.0. Fixes [#19944](#).
 - [renodr] - Update to libwacom-2.12.1. Fixes [#19938](#).
- June 11th, 2024
 - [renodr] - Update to Spidermonkey-115.12.0 (Security Update). Fixes [#19933](#).
 - [renodr] - Update to firefox-115.12.0esr (Security Update). Fixes [#19934](#).
 - [renodr] - Update to cups-2.4.9 (Security Update). Fixes [#19937](#).
 - [renodr] - Update to mesa-24.1.1. Includes adding the Ply python module, rust-bindgen, SPIRV-LLVM-Translator, and the libclc packages. Fixes [#19832](#).
 - [bdubbs] - Update to qemu-9.0.1. Fixes [#19926](#).
 - [bdubbs] - Update to glib-2.80.3. Fixes [#19932](#).
 - [bdubbs] - Update to libaom-3.9.1 (Security update). Fixes [#19935](#).
 - [renodr] - Update to xfce4-settings-4.18.6. Fixes [#19936](#).
 - [renodr] - Archive telepathy-mission-control since it is not needed by any other packages.
 - [bdubbs] - Update to btrfs-progs-v6.9. Fixes [#19915](#).
- June 10th, 2024
 - [bdubbs] - Update to pango-1.54.0. Fixes [#19928](#).
 - [bdubbs] - Update to packaging-24.1 (Python module). Fixes [#19927](#).
 - [timtas] - Update to xfce4-settings-4.18.5. Fixes [#19931](#).
 - [timtas] - Update to xfce4-session-4.18.4. Fixes [#19930](#).

- [timtas] - Update to xfce4-power-manager-4.18.4. Fixes [#19929](#).
- June 9th, 2024
 - [bdubbs] - Update to xscreensaver-6.09. Fixes [#19925](#).
 - [bdubbs] - Update to pciutils-3.13.0. Fixes [#19924](#).
- June 8th, 2024
 - [bdubbs] - Update to nss-3.101. Fixes [#19923](#).
 - [bdubbs] - Update to xkeyboard-config-2.42. Fixes [#19920](#).
 - [bdubbs] - Update to c-ares-1.30.0. Fixes [#19919](#).
 - [bdubbs] - Update to qpdf-11.9.1. Fixes [#19918](#).
 - [bdubbs] - Update to fribidi-1.0.15. Fixes [#19917](#).
 - [bdubbs] - Update to pcre2-10.44. Fixes [#19916](#).
- June 7th, 2024
 - [bdubbs] - Update Vulkan-Headers and Vulkan-Loader to version 1.3.287. Fixes [#19879](#).
 - [bdubbs] - Update to llvm-18.1.7. Fixes [#19903](#).
 - [bdubbs] - Update to mupdf-1.24.3. Fixes [#19914](#).
 - [bdubbs] - Update to icewm-3.5.1. Fixes [#19911](#).
 - [renodr] - Update to libreoffice-24.2.4.2 (Security Update). Fixes [#19912](#).
 - [renodr] - Update to php-8.3.8 (Security Update). Fixes [#19908](#).
 - [renodr] - Update to vlc-3.0.21 (Security Update). Fixes [#19913](#).
 - [renodr] - Update to libwacom-2.12.0. Fixes [#19906](#).
 - [renodr] - Update to cmake-3.29.5. Fixes [#19907](#).
- June 5th, 2024
 - [bdubbs] - Archive gstreamer-vaapi. Fixes [#19899](#).
 - [bdubbs] - Update to sentry_sdk-2.4.0 (Python module). Fixes [#19901](#).
 - [bdubbs] - Update to pytest-8.2.2 (Python module). Fixes [#19900](#).
 - [bdubbs] - Update to cmake-3.29.4. Fixes [#19898](#).
 - [bdubbs] - Update to poppler-24.06.0. Fixes [#19896](#).
 - [bdubbs] - Update to opencv and opencv_contrib-4.10.0. Fixes [#19895](#).
 - [renodr] - Update to vte-0.76.3 (Security Update). Fixes [#19897](#).
 - [renodr] - Update to wireplumber-0.5.3. Fixes [#19894](#).
 - [renodr] - Update to NetworkManager-1.48.0. Fixes [#19876](#).
 - [renodr] - Update to procmail-3.24. Fixes [#19891](#).
 - [renodr] - Fix building ncftp-3.2.7 with gcc-14. Fixes [#19889](#).
 - [renodr] - Fix building libgee-0.20.6 with gcc-14. Fixes [#19884](#).
- June 4th, 2024
 - [renodr] - Fix building telepathy-glib-0.24.2 with gcc14. Fixes [#19885](#).
 - [renodr] - Fix building gtksourceview-3.24.11 with gcc14. Fixes [#19883](#).
 - [renodr] - Fix building xine-lib with ffmpeg-7. Fixes [#19718](#).
- June 2nd, 2024
 - [bdubbs] - Update to logrotate-3.22.0. Fixes [#19892](#).
 - [bdubbs] - Update to highway-1.2.0. Fixes [#19881](#).
 - [bdubbs] - Update xorg-server tearfree patch for gcc13.
 - [bdubbs] - Update to libevdev-1.13.2. Fixes [#19878](#).
 - [bdubbs] - Update to libdrm-2.4.121. Fixes [#19893](#).
 - [thomas] - Fix a gcc14 issue in cyrus-sasl. Fixes [#19890](#).
- June 1st, 2024
 - [bdubbs] - Update gtk-doc dependencies. Fixes [#19880](#).
 - [bdubbs] - Update to lynx2.9.2. Fixes [#19877](#).
 - [bdubbs] - Update to hwdata-0.383. Fixes [#19874](#).
 - [bdubbs] - Update to git-2.45.2. Fixes [#19875](#).
 - [xry111] - Update to LLVM-18.1.6. Fixes [#19438](#).

- May 31st, 2024
 - [renodr] - Update to epiphany-46.1. Fixes [#19872](#).
 - [renodr] - Update to gtksourceview-5.12.1. Fixes [#19871](#).
 - [renodr] - Update to gucharmap-15.1.5. Fixes [#19856](#).
 - [renodr] - Update to gnome-terminal-3.52.2. Fixes [#19855](#).
 - [renodr] - Update to nautilus-46.2. Fixes [#19852](#).
 - [renodr] - Update to libshumate-1.2.2. Fixes [#19850](#).
 - [renodr] - Update to transmission-4.0.6. Fixes [#19867](#).
 - [bdubbs] - Update to libvpx-1.14.1. Fixes [#19873](#).
 - [bdubbs] - Update to ruby-3.3.2. Fixes [#19870](#).
 - [bdubbs] - Update to wayland-1.23.0. Fixes [#19869](#).
 - [bdubbs] - Update to libcap-2.70. Fixes [#19814](#).
 - [renodr] - Update to gdm-46.2. Fixes [#19868](#).
 - [renodr] - Update to gnome-control-center-46.2. Fixes [#19860](#).
 - [renodr] - Update to xdg-desktop-portal-gnome-46.2. Fixes [#19854](#).
 - [renodr] - Update to mutter-46.2. Fixes [#19851](#).
 - [renodr] - Update gnome-shell and gnome-shell-extensions to 46.2. Fixes [#19849](#).
 - [renodr] - Update to gnome-online-accounts-3.50.2. Fixes [#19848](#).
 - [renodr] - Update to evolution-data-server and evolution 3.52.2. Fixes [#19842](#).
 - [bdubbs] - Restore PyYAML.
- May 30th, 2024
 - [renodr] - Update to firefox-115.11.0esr (Security Update). Fixes [#19792](#).
 - [renodr] - Disable the sandbox in Firefox on i686 due to issues with system call filtering. Fixes [#19775](#).
 - [renodr] - Update to spidermonkey-115.11.0 (Security Update). Fixes [#19787](#).
 - [renodr] - Update to thunderbird-115.11.1 (Security Update). Fixes [#19798](#).
 - [bdubbs] - Update to gstreamer-1.24.4 and associated plugins. Fixes [#19864](#).
 - [bdubbs] - Update to mariadb-10.11.8 (Security Update). Fixes [#19865](#).
 - [bdubbs] - Update to requests-2.32.3 (Python module). Fixes [#19866](#).
- May 29th, 2024
 - [renodr] - Update to OpenJDK-22.0.1 (Security Update). Fixes [#19508](#).
 - [renodr] - Update to webkitgtk-2.44.2 (Security Update). Fixes [#19805](#).
 - [renodr] - Update to postgresql-16.3 (Security Update). Fixes [#19779](#).
 - [bdubbs] - Update to node-20.14.0. Fixes [#19862](#).
 - [bdubbs] - Add dolphin and dolphin-plugins to KDE applications.
 - [thomas] - Update to ISC Kea-2.6.0. Fixes [#19863](#).
- May 28th, 2024
 - [bdubbs] - Update to ffmpeg-7.0.1. Fixes [#19861](#).
 - [bdubbs] - Update to adwaita-icon-theme-46.2. Fixes [#19859](#).
 - [bdubbs] - Update to enchant-2.8.0. Fixes [#19857](#).
 - [bdubbs] - Update to gsl-2.8. Fixes [#19846](#).
 - [bdubbs] - Update to pipewire-1.0.7. Fixes [#19845](#).
- May 27th, 2024
 - [bdubbs] - Add a correction to xine-lib to handle ffmpeg-7. Fixes [#19718](#).
 - [bdubbs] - Add patches to vlc to handle ffmpeg-7 and gcc-14. Addresses [#19718](#).
 - [bdubbs] - Update to protobuf-27.0. Fixes [#19834](#).
 - [bdubbs] - Update to libadwaita-1.5.1. Fixes [#19841](#).
 - [bdubbs] - Update to librsvg-2.58.1. Fixes [#19838](#).
 - [bdubbs] - Update to plasma-wayland-protocols-1.13.0. Fixes [#19837](#).
 - [bdubbs] - Update to sentry_sdk-2.3.1 (Python module). Fixes [#19835](#).
 - [thomas] - Update to evince-46.3. Fixes [#19844](#).
- May 26th, 2024
 - [bdubbs] - Update to kf6 and plasma packages used by lxqt. Fixes [#19781](#).

- [bdubbs] - Update to plasma-6.0.5. Fixes [#19680](#).
 - [timtas] - Update to vte-0.76.2. Fixes [#19853](#).
 - [timtas] - Update to gvfs-1.54.1. Fixes [#19843](#).
 - [bdubbs] - Update to kde-gear-24.05.0 including falkon and kate. Fixes [#19833](#).
 - [thomas] - Update to ntp-4.2.8p18. Fixes [#19847](#).
- May 25th, 2024
 - [bdubbs] - Update to kf6-6.2.0 including extra-cmake-modules-6.2.0. Fixes [#19780](#).
 - [thomas] - Update to c-ares-1.29.0. Fixes [#19840](#).
 - [thomas] - Update to dhcpcd-10.0.8. Fixes [#19839](#).
 - [thomas] - Update to sqlite-3.46.0. Fixes [#19836](#).
- May 23rd, 2024
 - [bdubbs] - Update to xterm-392. Fixes [#19831](#).
 - [bdubbs] - Update to pavucontrol-6.0. Fixes [#19830](#).
 - [bdubbs] - Update to curl-8.8.0. Fixes [#19829](#).
 - [bdubbs] - Update to umockdev-0.18.3. Fixes [#19825](#).
 - [bdubbs] - Update to qt6-6.7.1 and qtwebengine-6.7.1. Fixes [#19822](#).
 - [bdubbs] - Update latest Intel microcode version (Security Update). Fixes [#19797](#).
- May 22nd, 2024
 - [renodr] - Update to SPIRV-Headers and SPIRV-Tools 1.3.283.0. Fixes [#19795](#).
 - [renodr] - Update Vulkan-Headers and Vulkan-Loader to 1.3.285. Fixes [#19760](#).
 - [renodr] - Update to gnome-maps-46.11. Fixes [#19785](#).
 - [renodr] - Update to file-roller-44.3. Fixes [#19770](#).
 - [renodr] - Update to snapshot-46.3. Fixes [#19759](#).
 - [renodr] - Update to gnome-calculator-46.1. Fixes [#19758](#).
 - [renodr] - Fix a regression in gnome-shell that appears when using glib-2.80.2.
 - [renodr] - Update to glib-2.80.2 (Security Update). Fixes [#19764](#).
 - [bdubbs] - Update to icewm-3.5.0. Fixes [#19824](#).
 - [bdubbs] - Update to hicolor-icon-theme-0.18. Fixes [#19823](#).
 - [bdubbs] - Update to libass-0.17.2. Fixes [#19817](#).
 - [rahul] - Update to nodejs-20.13.1. Fixes [#19765](#).
 - [rahul] - Update to mesa-24.0.8. Fixes [#19767](#).
 - [rahul] - Update to harfbuzz-8.5.0. Fixes [#19789](#).
 - [rahul] - Update to bluez-5.76. Fixes [#19806](#).
 - [rahul] - Update to pipewire-1.0.6. Fixes [#19782](#).
 - [bdubbs] - Update to requests-2.32.2 (Python module). Fixes [#19821](#).
 - [bdubbs] - Update to gi_docgen-2024.1 (Python module). Fixes [#19820](#).
 - [bdubbs] - Update to Mako-1.3.5 (Python module). Fixes [#19790](#).
 - [bdubbs] - Update to lxml-5.2.2 (Python module). Fixes [#19783](#).
 - [bdubbs] - Update to sentry_sdk-2.2.1 (Python module). Fixes [#19826](#).
 - [timtas] - Update to openldap-2.6.8. Fixes [#19827](#).
- May 21st, 2024
 - [bdubbs] - Update to doxygen-1.11.0. Fixes [#19819](#).
 - [bdubbs] - Update to nghttp2-1.62.1. Fixes [#19816](#).
 - [bdubbs] - Update to ghostscript-10.03.1 (Security Update). Fixes [#19813](#).
 - [bdubbs] - Update to pytest-8.2.1 (Python module). Fixes [#19815](#).
- May 20th, 2024
 - [bdubbs] - Update to gdk-pixbuf-2.42.12 (Security Update). Fixes [#19803](#).
 - [bdubbs] - Update to libxml2-2.12.7 (Security Update). Fixes [#19788](#).
 - [bdubbs] - Update to xfsprogs-6.8.0. Fixes [#19809](#).
 - [bdubbs] - Update to LVM2.2.03.24. Fixes [#19808](#).
- May 19th, 2024
 - [bdubbs] - Update to wireshark-4.2.5 (Security Update). Fixes [#19801](#).

- [bdubbs] - Update to xwayland-24.1.0. Fixes [#19802](#).
 - [bdubbs] - Update to unrar-7.0.9. Fixes [#19799](#).
- May 18th, 2024
 - [bdubbs] - Update to iw-6.9. Fixes [#19811](#).
 - [bdubbs] - Update to asciidoctor-2.0.23. Fixes [#19810](#).
 - [bdubbs] - Update to sentry_sdk-2.2.0 (Python module). Fixes [#19807](#).
 - [bdubbs] - Update to lxqt.openssh-askpass-2.0.1. Fixes [#19786](#).
 - [bdubbs] - Update to qtermwidget and qterminal-2.0.0. Fixes [#19812](#).
 - [bdubbs] - Update to kirigami-addons-1.2.1. Fixes [#19776](#).
 - [bdubbs] - Update to gcc-14.1. Fixes [#19762](#).
 - [bdubbs] - Update to lxqt-panel-2.0.1. Fixes [#19772](#).
 - [bdubbs] - Update to libfm-qt-2.0.2. Fixes [#19771](#).
 - [bdubbs] - Update to unbound-1.20.0. Fixes [#19769](#).
 - [timtas] - Update to gtk+-3-3.24.42. Fixes [#19804](#).
- May 16th, 2024
 - [thomas] - Update to mupdf-1.24.2. Fixes [#19773](#).
 - [thomas] - Update to libslirp-4.8.0. Fixes [#19778](#).
 - [thomas] - Update to nghttp2-1.62.0. Fixes [#19791](#).
 - [thomas] - Update to bind-9.18.27, bind-utils-9.18.27. Fixes [#19800](#).
- May 15th, 2024
 - [timtas] - Update to git-2.45.1. Fixes [#19796](#).
 - [timtas] - Update to gimp-2.10.38. Fixes [#19793](#).
 - [thomas] - Update to php-8.3.7. Fixes [#19777](#).
 - [thomas] - Update to xterm-391. Fixes [#19784](#).
- May 14th, 2024
 - [renodr] - Fix building Inkscape with poppler-24.05.0. Fixes [#19794](#).
 - [renodr] - Fix building gst-libav with ffmpeg-7.
- May 13th, 2024
 - [renodr] - Fix building sphinx_rtd_theme with docutils-0.21.x. This fixes an error about incompatible versions, but upstream has changed the version range so that it's compatible.
 - [timtas] - Update to samba-4.20.1. Fixes [#19768](#).
- May 11th, 2024
 - [thomas] - Update to nss-3.100. Fixes [#19763](#).
- May 10th, 2024
 - [thomas] - Update to cmake-3.29.3. Fixes [#19766](#).
- May 6th, 2024
 - [bdubbs] - Update to libqalculate-5.1.1. Fixes [#19757](#).
 - [bdubbs] - Update to sentry_sdk-2.1.1 (Python module). Fixes [#19757](#).
 - [bdubbs] - Update to mercurial-6.7.3. Fixes [#19756](#).
- May 6th, 2024
 - [bdubbs] - Update to lxml-5.2.1 (Python module). Fixes [#19068](#).
 - [bdubbs] - Update to sphinx-7.3.7 (Python module). Fixes [#19676](#).
 - [bdubbs] - Update to sphinxcontrib_applehelp-1.0.8 (Python module).
 - [bdubbs] - Update to alabaster-0.7.16 (Python module).
 - [bdubbs] - Update to Cython-3.0.10 (Python module). Fixes [#18303](#).
 - [bdubbs] - Archive PyYaml. Addresses [#18303](#).
- May 5th, 2024
 - [xry111] - Update to rustc-1.78.0. Fixes [#19557](#).
 - [bdubbs] - Update to libfm-qt-2.0.1, lximage-qt-2.0.1, and lxqt-notificationd-2.0.1. Fixes [#19753](#).
 - [bdubbs] - Update to xdg-desktop-portal-lxqt-1.0.2. Fixes [#19754](#).
 - [bdubbs] - Update to pygments-2.18.0 (Python module). Fixes [#19752](#).

- [bdubbs] - Update to enchant-2.7.3. Fixes [#19751](#).
 - [xry111] - Update Python dependencies: attrs-23.2.0, meson_python-0.16.0, and pyproject-metadata-0.8.0. Add hatch-fancy-pypi-readme-24.1.0 to support attrs-23.2.0. Addresses [#18562](#).
 - [xry111] - Update Python dependencies: editables-0.5, hatchling-1.24.2, hatch-vcs-0.4.0, pathspec-0.12.1, pluggy-1.5.0, setuptools_scm-8.0.4, and typing_extensions-4.11.0. Add trove-classifiers-2024.4.10 to support hatchling-1.24.2. Addresses [#18562](#).
- May 4th, 2024
 - [xry111] - Archive py which is no longer needed by pytest.
- May 3rd, 2024
 - [bdubbs] - Update to libnvme-1.9. Fixes [#19749](#).
 - [bdubbs] - Update to hodata-0.382. Fixes [#19750](#).
 - [bdubbs] - Update to libreoffice-24.2.3.2. Fixes [#19745](#).
 - [bdubbs] - Update to ibus-1.5.30. Fixes [#19747](#).
 - [bdubbs] - Update to lynx-2.9.1. Fixes [#19748](#).
 - [thomas] - Update to btrfs-progs-6.8.1. Fixes [#19742](#).
- May 2nd, 2024
 - [renodr] - Update to Net-DNS-1.45 (Perl Module). Fixes [#19743](#).
 - [renodr] - Update to glslang-14.2.0. Fixes [#19744](#).
 - [renodr] - Update to gtk4-4.14.4. Fixes [#19746](#).
 - [renodr] - Update to tracker and tracker-miners 3.7.3. Fixes [#19716](#).
 - [bdubbs] - Update to poppler-24.05.0. Fixes [#19741](#).
 - [bdubbs] - Update to nano-8.0. Fixes [#19740](#).
- May 1st, 2024
 - [timtas] - Update to cups-2.4.8. Fixes [#19729](#).
- April 30th, 2024
 - [bdubbs] - Update the gstreamer stack to 1.24.3. Fixes [#19737](#).
 - [renodr] - Update to tracker and tracker-miners 3.7.2. Fixes [#19716](#).
 - [renodr] - Update to gnome-tweaks-46.1. Fixes [#19728](#).
 - [thomas] - Update to git-2.45.0. Fixes [#19736](#).
 - [thomas] - Update to evince-46.1. Fixes [#19732](#).
- April 29th, 2024
 - [bdubbs] - Update to graphviz-11.0.0. Fixes [#19735](#).
 - [bdubbs] - Update to gedit-47.0. Fixes [#19734](#).
 - [bdubbs] - Update to sentry-sdk-2.0.1 (Python module). Fixes [#19723](#).
- April 28th, 2024
 - [bdubbs] - Update to mlt-7.24.0. Fixes [#19733](#).
 - [bdubbs] - Update to jasper-4.2.4. Fixes [#19731](#).
 - [bdubbs] - Update to libgedit-gtksourceview-299.2.1. Fixes [#19730](#).
- April 27th, 2024
 - [bdubbs] - Update to valgrind-3.23.0. Fixes [#19727](#).
 - [bdubbs] - Update to libarchive-3.7.4 (Security Update). Fixes [#19724](#).
 - [bdubbs] - Update to AppStream-1.0.3. Fixes [#19714](#).
 - [bdubbs] - Update to pytest-8.2.0 (Python module). Fixes [#19726](#).
 - [bdubbs] - Update to wayland-protocols-1.36. Fixes [#19725](#).
 - [bdubbs] - Update to unrar-7.0.8. Fixes [#19722](#).
 - [bdubbs] - Update to fribidi-1.0.14. Fixes [#19721](#).
 - [bdubbs] - Update to enchant-2.7.2. Fixes [#19719](#).
 - [bdubbs] - Update to libgpg-error-1.49. Fixes [#19720](#).
 - [bdubbs] - Update to ed-1.20.2. Fixes [#19713](#).
 - [bdubbs] - Update to libaom-3.9.0. Fixes [#19712](#).
- April 26th, 2024
 - [bdubbs] - Update to docutils-0.21.2 (Python module). Fixes [#19710](#).

- April 25th, 2024
 - [timtas] - Update to mesa-24.0.6. Fixes [#19715](#).
 - [timtas] - Update to qemu-9.0.0. Fixes [#19517](#).
 - [xry111] - Patch SeaMonkey to unbreak building with ICU-75.1.
- April 24th, 2024
 - [renodr] - Update to ruby-3.3.1 (Security Update). Fixes [#19711](#).
- April 23rd, 2024
 - [renodr] - Update to gnome-control-center-46.1. Fixes [#19708](#).
 - [renodr] - Update to xdg-desktop-portal-gnome-46.1. Fixes [#19702](#).
 - [renodr] - Update to nautilus-46.1. Fixes [#19701](#).
 - [renodr] - Update to mutter-46.1. Fixes [#19700](#).
 - [renodr] - Update to gnome-user-docs-46.1. Fixes [#19699](#).
 - [renodr] - Update to gnome-shell-extensions-46.1. Fixes [#19698](#).
 - [renodr] - Update to gnome-shell-46.1. Fixes [#19698](#).
 - [renodr] - Update to gucharmap-15.1.4. Fixes [#19696](#).
 - [renodr] - Update to wireplumber-0.5.2. Fixes [#19709](#).
 - [renodr] - Update to gnome-terminal-3.52.1. Fixes [#19695](#).
 - [renodr] - Update to gnome-maps-46.10. Fixes [#19694](#).
 - [renodr] - Update to vte-0.76.1. Fixes [#19693](#).
 - [renodr] - Update to libshumate-1.2.1. Fixes [#19692](#).
 - [renodr] - Update to evolution-3.52.1. Fixes [#19690](#).
 - [renodr] - Update to evolution-data-server-3.52.1. Fixes [#19690](#).
 - [renodr] - Update Vulkan-Headers and Vulkan-Loader to 1.3.283. Fixes [#19689](#).
 - [bdubbs] - Update to nmap-7.95. Fixes [#19707](#).
 - [bdubbs] - Update to libxmlb-0.3.19. Fixes [#19706](#).
 - [bdubbs] - Update to libgusb-0.4.9. Fixes [#19705](#).
- April 22nd, 2024
 - [bdubbs] - Update to solid-6.1.1. Fixes [#19703](#).
 - [bdubbs] - Update to enchant-2.7.0. Fixes [#19697](#).
 - [bdubbs] - Update to gdk-pixbuf-2.42.11. Fixes [#19691](#).
 - [bdubbs] - Update to vala-0.56.17. Fixes [#19688](#).
- April 21st, 2024
 - [bdubbs] - Update to lxqt-2.0. Fixes [#19681](#) and [#19687](#).
- April 20th, 2024
 - [rahul] - Update to ffmpeg-7.0. Fixes [#19604](#).
 - [rahul] - Update to nodejs-20.12.2. Fixes [#19638](#).
 - [rahul] - Update to mesa-24.0.5. Fixes [#19641](#).
 - [rahul] - Update to power-profiles-daemon-0.21. Fixes [#19581](#).
- April 19th, 2024
 - [renodr] - Update to thunderbird-115.10.1. Fixes [#19685](#).
 - [renodr] - Update to gtk-4.14.3. Fixes [#19686](#).
- April 18th, 2024
 - [renodr] - Update to mpv-0.38.0. Fixes [#19682](#).
 - [bdubbs] - Update to icu-75.1. Fixes [#19674](#).
 - [bdubbs] - Update to wayland-protocols-1.35. Fixes [#19675](#).
 - [bdubbs] - Update to glibmm-2.66.7. Fixes [#19673](#).
 - [thomas] - Update to bind-9.18.26, bind-utils-9.18.26. Fixes [#19679](#).
 - [thomas] - Update to xfsprogs-6.7.0. Fixes [#19677](#).
 - [thomas] - Update to NASM-2.16.03. Fixes [#19678](#).
- April 17th, 2024
 - [bdubbs] - Update to Vulkan-Loader-1.3.282. Fixes [#19670](#).

- [bdubbs] - Update to util-macros-1.20.1. Fixes [#19669](#).
 - [bdubbs] - Update to libXmu-1.2.1 (Xorg library). Fixes [#19668](#).
 - [renodr] - Update to thunderbird-115.10.0 (Security Update). Fixes [#19671](#).
 - [renodr] - Update to firefox-115.10.0esr (Security Update). Fixes [#19664](#).
 - [renodr] - Update to spidermonkey-115.10.0 (Security Update). Fixes [#19666](#).
 - [renodr] - Update to pipewire-1.0.5. Fixes [#19665](#).
- April 16th, 2024
 - [bdubbs] - Update to php-8.3.6 (security update). Fixes [#19645](#).
 - [bdubbs] - Update to sqlite-autoconf-3450300 (3.45.3). Fixes [#19662](#).
 - [bdubbs] - Update to boost-1.85.0. Fixes [#19660](#).
 - [ken] - Change details for KDE in tuning fontconfig and note much earlier that the settings for fontconfig may be ignored by applications and desktop environments. Fixes [#19667](#).
 - [renodr] - Update to xf86-input-wacom-1.2.2. Fixes [#19663](#).
 - [renodr] - Update to libwacom-2.11.0. Fixes [#19661](#).
 - [renodr] - Update to gnome-system-monitor-46.0. Fixes [#19606](#).
 - [renodr] - Add gtkmm-4.14.0 to the book. Fixes [#14443](#).
 - [renodr] - Addatkmm-2.36.3 to the book. Fixes [#14406](#).
 - [renodr] - Add pangomm-2.52.0 to the book. Fixes [#14405](#).
 - [renodr] - Add cairomm-1.18.0 to the book. Fixes [#14172](#).
 - [renodr] - Add glibmm-2.80.0 to the book. Fixes [#14403](#).
 - [renodr] - Add libsigc++3 to the book. Fixes [#16086](#).
 - [bdubbs] - Update to bluez-5.75. Fixes [#19655](#).
 - [bdubbs] - Fix a build problem for sphinx-7.2.6. Fixes [#19659](#).
 - [bdubbs] - Update to elogind-255.4-r2. Fixes [#19298](#).
- April 15th, 2024
 - [renodr] - Update to gucharmap-15.1.3. Fixes [#19477](#).
 - [renodr] - Update to gnome-terminal-3.52.0. Fixes [#19476](#).
 - [renodr] - Update to gnome-logs-45.0. Fixes [#19603](#).
 - [renodr] - Update to gnome-disk-utility-46.0. Fixes [#19602](#).
 - [renodr] - Update to gnome-calculator-46.0. Fixes [#19599](#).
 - [renodr] - Update to file-roller-44.1. Fixes [#19598](#).
 - [renodr] - Update to evince-46.0. Fixes [#19490](#).
 - [renodr] - Update to dvisvgm-3.3. Fixes [#19642](#).
 - [renodr] - Update to asymptote-2.89. Fixes [#19543](#).
 - [renodr] - Fix bugs in latex2e and dvipdfm-x in texlive. Fixes [#19571](#).
 - [renodr] - Update to EOG-45.3. Fixes [#19475](#).
 - [renodr] - Update to baobab-46.0. Fixes [#19601](#).
 - [renodr] - Update to simple-scan-46.0. Fixes [#19620](#).
 - [bdubbs] - Update to Python3-3.12.3. Fixes [#19633](#).
 - [renodr] - Fix CVE-2024-25081 and CVE-2024-25082 in FontForge. Fixes [#19545](#).
 - [renodr] - Update to libxcb-1.17.0. Fixes [#19658](#).
 - [renodr] - Update to xcbproto-1.17.0. Fixes [#19657](#).
 - [renodr] - Update to vulkan-headers-1.3.282. Fixes [#19656](#).
 - [renodr] - Update to gtk-4.14.2. Fixes [#19583](#).
- April 14th, 2024
 - [bdubbs] - Update the gstreamer stack to 1.24.2. Fixes [#19634](#).
 - [bdubbs] - Remove overwriting of terminfo data in xterm. Fixes [#19611](#).
 - [bdubbs] - Update to lxqt-qtplugin-1.4.1. Fixes [#19652](#).
 - [bdubbs] - Update to plasma-6.0.3. Fixes [#19733](#).
 - [thomas] - Update to libwebp-1.4.0. Fixes [#19654](#).
 - [thomas] - Update to opus-1.5.2. Fixes [#19653](#).
- April 12th, 2024
 - [bdubbs] - Update to kf6-6.1.0. Fixes [#19649](#).

- April 12th, 2024
 - [renodr] - Update to gnome-user-docs-46.0. Fixes [#19497](#).
 - [renodr] - Update to gnome-tweaks-46.0. Fixes [#19496](#).
 - [renodr] - Update to gnome-session-46.0. Fixes [#19504](#).
 - [renodr] - Update to gnome-shell-extensions-46.0. Fixes [#19472](#).
 - [bdubbs] - Update to cmake-3.29.2. Fixes [#19644](#).
 - [bdubbs] - Update to cryptsetup-2.7.2. Fixes [#19630](#).
 - [renodr] - Update to gdm-46.0. Fixes [#19505](#).
 - [renodr] - Update to gnome-shell-46.0. Fixes [#19472](#).
 - [renodr] - Update to mutter-46.0. Fixes [#19473](#).
 - [renodr] - Update to gnome-control-center-46.0.1. Fixes [#19502](#).
 - [renodr] - Update to tecla-46.0. Fixes [#19486](#).
 - [renodr] - Update to gnome-settings-daemon-46.0. Fixes [#19485](#).
 - [renodr] - Update to gnome-bluetooth-46.0. Fixes [#19503](#).
 - [renodr] - Update to nautilus-46.0. Fixes [#19501](#).
 - [bdubbs] - Update to libxmlb-0.3.18. Fixes [#19631](#).
 - [bdubbs] - Update to taglib-2.0.1. Fixes [#19635](#).
 - [bdubbs] - Update to sentry-sdk-1.45.0 (Python module). Fixes [#19640](#).
 - [bdubbs] - Update to xorg-server-21.1.13. Fixes [#19650](#).
 - [bdubbs] - Update to xwayland-23.2.6. Fixes [#19623](#).
 - [renodr] - Update to snapshot-46.2. Fixes [#19648](#).
 - [renodr] - Update to gnome-online-accounts-3.50.1. Fixes [#19647](#).
 - [renodr] - Update to gcr4-4.3.0. Fixes [#19646](#).
 - [renodr] - Update to libgtop-2.41.3. Fixes [#19627](#).
 - [renodr] - Update to epiphany-46.0. Fixes [#19491](#).
 - [renodr] - Update to WebKitGTK-2.44.1. Fixes [#19622](#).
 - [bdubbs] - Update to docutils-0.21.1 (Python module). Fixes [#19632](#).
 - [bdubbs] - Update to Mako-1.3.3 (Python module). Fixes [#19639](#).
- April 11th, 2024
 - [renodr] - Add libjxl to the book in support gnome-backgrounds and other packages. Fixes [#19626](#).
 - [renodr] - Add highway to the book in support of libjxl. Fixes [#19626](#).
 - [thomas] - Update to Linux-PAM-1.6.1. Fixes [#19629](#).
 - [bdubbs] - Update to seamonkey-2.53.18.2. Fixes [#19617](#).
- April 10th, 2024
 - [bdubbs] - Update to upower-v1.90.4. Fixes [#19621](#).
 - [bdubbs] - Update to libarchive-3.7.3. Fixes [#19618](#).
 - [bdubbs] - Update to js-115.9.1 (spidermonkey). Fixes [#19616](#).
 - [timtas] - Update to rsync-3.3.0. Fixes [#19614](#).
 - [xry111] - Patch pipewire-1.0.4 to fix an issue breaking snapshot-46.1. Fixes [#19637](#).
- April 9th, 2024
 - [renodr] - Update to xdg-desktop-portal-gnome-46.0. Fixes [#19624](#).
 - [bdubbs] - Update to qt6-6.7.0 and qtwebengine-6.7.0. Fixes [#19575](#).
 - [renodr] - Update to gnome-backgrounds-46.0. Fixes [#19625](#).
 - [renodr] - Update to tracker-miners-3.7.1. Fixes [#19482](#).
 - [renodr] - Update to tracker-3.7.1. Fixes [#19482](#).
 - [renodr] - Update to evolution-3.52.0. Fixes [#19481](#).
 - [renodr] - Update to evolution-data-server-3.52.0. Fixes [#19481](#).
 - [renodr] - Update to gnome-maps-46.0. Fixes [#19506](#).
 - [renodr] - Update to libshumate-1.2.0. Fixes [#19492](#).
 - [renodr] - Add abseil-cpp, protobuf, and protobuf-c in support of libshumate and other packages. Part of [#19492](#).
- April 8th, 2024
 - [renodr] - Update TigerVNC to use xorg-server-21.1.12. This protects Xvnc against the security vulnerabilities fixed in that update.

- [bdubbs] - Update to httpd-2.4.59 (Security Update). Fixes [#19507](#).
 - [renodr] - Update to gnome-online-accounts-3.50.0. Fixes [#19480](#).
 - [renodr] - Update to pygobject-3.48.2 (Python Module). Fixes [#19613](#).
 - [renodr] - Archive gnome-video-effects. Fixes [#19488](#).
 - [renodr] - Update to gtksourceview-5.12.0. Fixes [#19590](#).
 - [renodr] - Update to gnome-keyring-46.1. Fixes [#19484](#).
 - [bdubbs] - Update to xwayland-23.2.5 (Security Update). Fixes [#19579](#).
 - [bdubbs] - Update to xorg-server-21.1.12 (Security Update). Addresses [#19579](#).
 - [bdubbs] - Update to nghttp2-1.61.0 (Security Update). Fixes [#19596](#).
- April 7th, 2024
 - [bdubbs] - Update to libX11-1.8.9 (Xorg library). Fixes [#19610](#).
 - [bdubbs] - Update to mtddev-1.1.7. Fixes [#19608](#).
 - [bdubbs] - Update to tcsh-6.24.12. Fixes [#19607](#).
 - [bdubbs] - Update to tepl-6.9.0. Fixes [#19591](#).
 - [bdubbs] - Update to libgedit-gtksourceview-299.1.0. Fixes [#19515](#).
 - [rahul] - Update to nodejs-20.12.1 (Security Update). Fixes [#19552](#).
 - [rahul] - Update to samba-4.20.0 (Security Update). Fixes [#19554](#).
 - [rahul] - Update to mesa-24.0.4. Fixes [#19556](#).
- April 6th, 2024
 - [bdubbs] - Move luit from xorg apps to it's own page. Fixes [#19578](#).
 - [bdubbs] - Update to gnutls-3.8.5. Fixes [#19592](#).
 - [bdubbs] - Update to enchant-2.6.9. Fixes [#19593](#).
 - [bdubbs] - Update to hwdata-0.381. Fixes [#19594](#).
 - [bdubbs] - Update to at-spi2-core-2.52.0. Fixes [#19587](#).
 - [bdubbs] - Update to sentry-sdk-1.44.1. Fixes [#19582](#).
 - [bdubbs] - Move libwnck from gnome section to xfce section. Fixes [#19577](#).
 - [thomas] - Update to pciutils-3.12.0. Fixes [#19609](#).
- April 5th, 2024
 - [bdubbs] - Update to libxmlb-0.3.17. Fixes [#19580](#).
 - [thomas] - Update to gtkmm-3.24.9. Fixes [#19589](#).
 - [thomas] - Update to cairomm-1.14.5. Fixes [#19588](#).
 - [thomas] - Update to poppler-24.04.0. Fixes [#19574](#).
 - [thomas] - Update to mupdf-1.24.1. Fixes [#19576](#).
 - [thomas] - Update to mpg123-1.32.6. Fixes [#19605](#).
 - [thomas] - Update to nasm-2.16.02. Fixes [#19595](#).
 - [xry111] - Update to LLVM-18.1.2. Addresses [#19438](#).
- April 4th, 2024
 - [renodr] - Update to gnome-weather-46.0. Fixes [#19586](#).
 - [renodr] - Update to libgweather-4.4.2. Fixes [#19522](#).
 - [renodr] - Adapt Firefox to use Google's Location Service by removing our MLS API key. Fixes [#19541](#).
 - [renodr] - Adapt Geoclue to use Google's Location Service due to the Mozilla Location Service shutdown. Fixes [#19541](#).
 - [renodr] - Update to gjs-1.80.2. Fixes [#19487](#).
 - [renodr] - Update to snapshot-46.1. Fixes [#19507](#).
 - [renodr] - Update to wireplumber-0.5.1. Fixes [#19567](#).
 - [thomas] - Update to pango-1.52.2. Fixes [#19585](#).
 - [thomas] - Update to cmake-3.29.1. Fixes [#19584](#).
- April 3rd, 2024
 - [renodr] - Update to libadwaita-1.5.0. Fixes [#19489](#).
 - [renodr] - Update the gstreamer stack to 1.24.1. Fixes [#19408](#).
- April 2nd, 2024
 - [bdubbs] - Update to libreoffice-24.2.2.2. Fixes [#19559](#).

- [bdubbs] - Update to kirigami-addons-1.1.0. Fixes [#19573](#).
 - [bdubbs] - Update to gobject-introspection-1.80.1. Fixes [#19572](#).
 - [ken] - Update to biber-2.20 with biblatex-3.20 and update related perl items: Module-Build-0.4234 (Perl Module), Alien-Build-2.80, B-Hooks-EndOfScope-0.28, CPAN-Meta-Check-0.18, DateTime-1.65, DateTime-Locale-1.40, DateTime-TimeZone=2.62, Devel-StackTrace-2.05, Exporter-Tiny-1.006002, File-Listing-6.16, HTML-Tagset-3.24, HTTP-Cookies-6.11, HTTP-Date-6.06, HTTP-Message-6.45, Net-SSLeay-1.94, Test-Warnings-0.033, Tie-Cycle-1.228, Variable-Magic-0.64, XML-LibXML-2.0210 (Perl Dependent Modules). Archive: Importer, Module-Pluggable, Sub-Info (Perl Dependent Modules). Thanks to Stephen Berman for reporting Net-SSLeay no-longer passed its tests, and to Bruce for diagnosing this. Fixes [#19528](#).
- April 1st, 2024
 - [thomas] - Upgrade c-ares-1.28.1. Fixes [#19566](#).
 - [renodr] - Fix building Inkscape with poppler-24.03.0. Fixes [#19570](#).
 - [bdubbs] - Update to libical-3.0.18. Fixes [#19568](#).
 - [bdubbs] - Update to jasper-4.2.3. Fixes [#19565](#).
 - [bdubbs] - Update to soundtouch-2.3.3. Fixes [#19564](#).
 - [bdubbs] - Update to mercurial-6.7.2. Fixes [#19562](#).
 - [bdubbs] - Update to harfbuzz-8.4.0. Fixes [#19561](#).
- March 31st, 2024
 - [renodr] - Update to librsvg-2.58.0. Fixes [#19433](#).
- March 30th, 2024
 - [bdubbs] - Update to wireshark-4.2.4 (Security update). Fixes [#19555](#).
 - [bdubbs] - Update to sentry-sdk-1.44.0 (Python module). Fixes [#19558](#).
 - [thomas] - Update to c-ares-1.28.0. Fixes [#19563](#).
- March 29th, 2024
 - [thomas] - Update to shadow-4.15.1. Fixes [#19532](#).
- March 28th, 2024
 - [bdubbs] - Update to qt6-6.6.3 and qtwebengine-6.6.3. Fixes [#19551](#).
- March 28th, 2024
 - [timtas] - Update to libva-2.21.0. Fixes [#19546](#).
 - [bdubbs] - Update to bubblewrap-0.9.0. Fixes [#19549](#).
 - [bdubbs] - Update to URI-5.28 (Perl Module). Fixes [#19550](#).
 - [bdubbs] - Update to libblockdev-3.1.1. Fixes [#19548](#).
 - [bdubbs] - Update to btrfs-progs-v6.8. Fixes [#19547](#).
 - [bdubbs] - Update to xorgproto-2024.1. Fixes [#19544](#).
- March 27th, 2024
 - [timtas] - Update to cURL-8.7.1 (Security Update). Fixes [#19553](#).
 - [thomas] - Fix a misconfiguration in LibreOffice on i686 systems.
 - [timtas] - Force vlc to compile against lua52.
- March 26th, 2024
 - [renodr] - Update to Vulkan-Headers and Vulkan-Loader 1.3.281. Fixes [#19434](#).
 - [bdubbs] - Update to icewm-3.4.7. Fixes [#19542](#).
 - [bdubbs] - Update to Xorg libraries libX11-1.8.8 and libXmu-1.2.0. Fixes [#19539](#).
 - [bdubbs] - Update to emacs-29.3 (Security Update). Fixes [#19537](#).
- March 25th, 2024
 - [timtas] - Update to cryptsetup-2.7.1. Fixes [#19425](#).
 - [renodr] - Update to gsettings-desktop-schemas-46.0. Fixes [#19479](#).
- March 24th, 2024
 - [bdubbs] - Update to umockdev-0.18.1. Fixes [#19538](#).
 - [bdubbs] - Added Python modules certifi, psutil, pygdbmi, and sentry-sdk in support of plasma. Fixes [#19536](#).
 - [bdubbs] - Added Python modules html5lib and webencodings in support of qtwebengine. Fixes [#19535](#).
- March 24th, 2024
 - [bdubbs] - Update to libpciaccess-0.18.1 (xorg library). Fixes [#19534](#).

- [bdubbs] - Update to libxkbcommon-1.7.0. Fixes [#19533](#).
 - [bdubbs] - Update to SPIRV-Tools-1.3.280.0. Fixes [#19531](#).
 - [bdubbs] - Update to enchant-2.6.8. Fixes [#19530](#).
- March 23rd, 2024
 - [thomas] - Update to gnutls-3.8.4 (Security Update). Fixes [#19510](#).
 - [xry111] - Update to rustc-1.77.0. Fixes [#19527](#).
- March 22nd, 2024
 - [rahul] - Update to cmake-3.29.0. Fixes [#19525](#).
 - [rahul] - Update to gtk4-4.14.1. Fixes [#19464](#).
 - [rahul] - Update to pipewire-1.0.4. Fixes [#19462](#).
 - [renodr] - Update to spidermonkey-115.9.1 (Security Update). Fixes [#19500](#).
 - [ken] - Update to firefox-115.9.1 (Security Update). Fixes [#19529](#).
 - [bdubbs] - Update to mercurial-6.7.1. Fixes [#19526](#).
- March 21st, 2024
 - [bdubbs] - Update to bind utilities/bind-9.18.25. Fixes [#19521](#).
 - [bdubbs] - Update to wayland-protocols-1.34. Fixes [#19520](#).
 - [bdubbs] - Update to libcloudproviders-0.3.6. Fixes [#19519](#).
 - [bdubbs] - Update to SPIRV-Headers-1.3.280.0. Fixes [#19518](#).
 - [bdubbs] - Update to mupdf-1.24.0. Fixes [#19516](#).
 - [bdubbs] - Update to adwaita-icon-theme-46.0. Fixes [#19514](#).
 - [bdubbs] - Update to pinentry-1.3.0. Fixes [#19513](#).
 - [bdubbs] - Update to glad-2.0.6. Fixes [#19512](#).
- March 20th, 2024
 - [bdubbs] - Update to elogind-252.23. Fixes [#19509](#).
 - [bdubbs] - Update to harfbuzz-8.3.1. Fixes [#19494](#).
 - [bdubbs] - Update to mercurial-6.7. Fixes [#19469](#).
 - [ken] - Update to texlive 2024. Fixes [#19463](#).
 - [ken] - Remove old ConTeXt fixes from texlive source. Fixes [#18349](#).
 - [timtas] - Update to vte-0.76.0. Fixes [#19474](#).
 - [renodr] - Adapt rsync to LZ4 now being in LFS.
 - [bdubbs] - Update to bluefish-2.2.15. Fixes [#19493](#).
 - [bdubbs] - Update to wireplumber-0.5.0. Fixes [#19089](#).
 - [timtas] - Update to thunderbird-115.9.0. Fixes [#19515](#).
 - [thomas] - Update to libpaper-2.2.5. Fixes [#19511](#).
- March 19th, 2024
 - [bdubbs] - Update to tk8.6.14. Fixes [#19498](#).
 - [bdubbs] - Update to SCons-4.7.0. Fixes [#19495](#).
 - [bdubbs] - Update to glib-networking-2.80.0. Fixes [#19470](#).
 - [ken] - Update to firefox-115.9.0 (Security Update). Fixes [#19499](#).
 - [bdubbs] - Update to nss-3.99. Fixes [#19467](#).
 - [bdubbs] - Update to vala-0.56.16. Fixes [#19465](#).
 - [bdubbs] - Update to libaom-3.8.2. Fixes [#19461](#).
 - [thomas] - Update to libpaper-2.2.3. Fixes [#19445](#).
 - [bdubbs] - Update to tcsh-6.24.11. Fixes [#19457](#).
 - [bdubbs] - Update to jasper-4.2.2. Fixes [#19454](#).
 - [bdubbs] - Update to libqalculate-5.0.0. Fixes [#19453](#).
- March 18th, 2024
 - [bdubbs] - Update to xapian-core-1.4.25. Fixes [#19427](#).
 - [bdubbs] - Update to iceauth-1.0.10 (Xorg app). Fixes [#19450](#).
 - [bdubbs] - Update to libXaw-1.0.16 (Xorg library). Fixes [#19451](#).
 - [bdubbs] - Update to packaging-24.0 (Python module). Fixes [#19448](#).
 - [bdubbs] - Update to pygobject3-3.48.1 (Python module). Fixes [#19440](#).

- [bdubbs] - Update to pytest-8.1.1 (Python module). Fixes [#19443](#).
 - [timtas] - Update to gvfs-1.54.0. Fixes [#19483](#).
 - [bdubbs] - Update to HTML-Parser-3.82 (Perl module). Fixes [#19458](#).
 - [bdubbs] - Update to libwww-perl-6.77 (Perl module). Fixes [#19447](#).
 - [bdubbs] - Update to LWP-Protocol-https-6.14 (Perl module). Fixes [#19446](#).
 - [bdubbs] - Update to asciidoctor-2.0.22. Fixes [#19437](#).
 - [bdubbs] - Update to shadow-4.15.0. Fixes [#19432](#).
 - [thomas] - Update to glslang-14.1.0. Fixes [#19435](#).
 - [thomas] - Update to php-8.3.4. Fixes [#19466](#).
 - [thomas] - Update to wget-1.24.5. Fixes [#19449](#).
 - [thomas] - Update to at-spi2-core-2.50.2. Fixes [#19471](#).
- March 17th, 2024
 - [bdubbs] - Update to sddm-0.21.0. Fixes [#19360](#).
 - [bdubbs] - Add xdotool-3.20211022.1 in support of plasma6. Addresses [#19373](#).
 - [bdubbs] - Add libdisplay-info-0.1.1 in support of plasma6. Addresses [#19373](#).
 - [bdubbs] - Add hwdata-0.380 in support of plasma6. Addresses [#19373](#).
 - [bdubbs] - Add kirigami-addons-1.0.1 in support of plasma6. Addresses [#19373](#).
 - [bdubbs] - Add qcoro-0.10.0 in support of plasma6. Addresses [#19373](#).
 - [thomas] - Update to libxml2-2.12.6. Fixes [#19468](#).
 - [xry111] - Archive wpebackend-fdo and libwpe. Fixes [#18460](#).
 - [xry111] - Update to WebKitGTK-2.44.0. Fixes [#19478](#).
- March 16th, 2024
 - [timtas] - Update to mesa-24.0.3. Fixes [#19459](#).
- March 15th, 2024
 - [rahul] - Update to icewm-3.4.6. Fixes [#19430](#).
 - [rahul] - Update to unbound-1.19.3 (Security Update). Fixes [#19429](#).
 - [rahul] - Update to bluez-5.73. Fixes [#19428](#).
 - [rahul] - Update to gnupg-2.4.5. Fixes [#19426](#).
 - [thomas] - Upgrade to sqlite-3.45.2. Fixes [#19456](#).
- March 14th, 2024
 - [xry111] - Add dtc-1.7.0 for supporting qemu-8.2.2.
- March 12th, 2024
 - [ken] - Update to mutt-2.2.13. Fixes [#19441](#).
 - [thomas] - Update to openssh-9.7p1, ssh-askpass-9.7p1. Fixes [#19452](#).
- March 10th, 2024
 - [ken] - Update to asymptote-2.88. Fixes [#19372](#).
 - [xry111] - Update to glib-2.80.0. Fixes [#19444](#).
 - [xry111] - Update to gobject-introspection-1.80.0. Fixes [#19439](#).
 - [xry111] - Combine gobject-introspection into glib page to better handle the circular dependency between these two packages.
- March 9th, 2024
 - [ken] - Update to dvisvgm-3.2.2. Fixes [#19384](#).
 - [ken] - Update to ghostscript-10.03.0 (Security Update). Fixes [#19423](#).
 - [thomas] - Upgrade to postfix-3.9.0. Fixes [#19436](#).
 - [thomas] - Upgrade to libxfc4ui-4.18.6. Fixes [#19436](#).
- March 7th, 2024
 - [bdubbs] - Revert to pytest-8.0.2 (Python Module). Fixes [#19417](#).
 - [renodr] - Update to pyparsing-3.1.2 (Python Module). Fixes [#19416](#).
 - [renodr] - Update to libassuan-2.5.7. Fixes [#19415](#).
 - [renodr] - Update to SDL2-2.30.1. Fixes [#19412](#).
 - [renodr] - Update to umockdev-0.18.0. Fixes [#19399](#).
 - [renodr] - Update to opus-1.5.1. Fixes [#19409](#).

- [renodr] - Update to uhttplibmock-0.10.0. Fixes [#19406](#).
 - [renodr] - Update to vala-0.56.15. Fixes [#19405](#).
 - [renodr] - Update to mkfontscale-1.2.3, xauth-1.1.3, xev-1.2.6, xmmessage-1.0.7, xpr-1.2.0, and xrefresh-1.1.0 (Xorg Applications). Fixes [#19402](#).
 - [renodr] - Update to gdb-14.2. Fixes [#19398](#).
- March 6th, 2024
 - [xry111] - Update to SeaMonkey-2.53.18.1 (Security Update). Fixes [#19420](#).
 - [xry111] - Update to LLVM-18.1.0. Fixes [#19413](#).
 - [renodr] - Update to thunderbird-115.8.1 (Security Update). Fixes [#19411](#).
 - [renodr] - Update to xf86-input-wacom-1.2.1 (Xorg Driver). Fixes [#19403](#).
 - [renodr] - Update to gtk-doc-1.34.0. Fixes [#19410](#).
 - [renodr] - Update to gnome-maps-45.5. Fixes [#19397](#).
 - [renodr] - Update to gcr-4.2.1. Fixes [#19396](#).
 - [renodr] - Update to libadwaita-1.4.4. Fixes [#19395](#).
 - [renodr] - Update to pytest-8.1.0 (Python Module). Fixes [#19401](#).
 - [bdubbs] - Finish updating to kf6-apps. Fixes [#19375](#).
 - [xry111] - Archive PCRE1. Fixes [#18893](#).
- March 5th, 2024
 - [renodr] - Update to libreoffice-24.2.1.2. Fixes [#19382](#).
 - [timtas] - Update to qemu-8.2.2. Fixes [#19404](#).
- March 4th, 2024
 - [bdubbs] - Update to poppler-24.03.0. Fixes [#19400](#).
- March 4th, 2024
 - [bdubbs] - Update to kImageAnnotator-0.7.1. Fixes [#19388](#).
 - [bdubbs] - Update to kColorPicker-0.3.1. Fixes [#19387](#).
 - [bdubbs] - Preliminary update to kf6-6.0.0.
- March 3rd, 2024
 - [bdubbs] - Update to encodings-1.1.0 (Xorg Font). Fixes [#19393](#).
 - [bdubbs] - Update to libXcursor-1.2.2 (Xorg Library) and libfontenc-1.1.8 (Xorg Library). Fixes [#19392](#) and [#19389](#).
 - [bdubbs] - Update to libxcb-1.16.1. Fixes [#19391](#).
 - [bdubbs] - Update to libXdmc-1.1.5. Fixes [#19394](#).
 - [bdubbs] - Update to nghttp2-1.60.0. Fixes [#19386](#).
 - [bdubbs] - Update to mdadm-4.3. Fixes [#19377](#).
 - [bdubbs] - Update to pixman-0.43.4. Fixes [#19376](#).
 - [bdubbs] - Update to mesa-24.0.2. Fixes [#19374](#).
 - [bdubbs] - Update to a52dec-0.8.0. Fixes [#19368](#).
 - [bdubbs] - Update to swig-4.2.1. Fixes [#19365](#).
 - [bdubbs] - Update to qpdf-11.9.0. Fixes [#19363](#).
 - [bdubbs] - Update to libunistring-1.2. Fixes [#19361](#).
 - [bdubbs] - Update to libpng-1.6.43. Fixes [#19354](#).
 - [bdubbs] - Update to mupdf-1.23.11. Fixes [#19347](#).
 - [bdubbs] - Update to polkit-qt-1-0.200.0. Fixes [#19345](#).
 - [bdubbs] - Update to python-dbusmock-0.31.1 (Python module). Fixes [#19356](#).
 - [bdubbs] - Update to npth-1.7. Fixes [#19353](#).
 - [bdubbs] - Update to libksba-1.6.6. Fixes [#19352](#).
 - [bdubbs] - Update to libgpg-error-1.48. Fixes [#19351](#).
 - [thomas] - Update to pciutils-3.11.1. Fixes [#19364](#).
 - [xry111] - Update to shadow-4.14.6. Fixes [#19385](#).
- March 2nd, 2024
 - [renodr] - Update to epiphany-45.3. Fixes [#19381](#).
 - [renodr] - Update to glib-networking-2.78.1. Fixes [#19378](#).
 - [renodr] - Update to AppStream-1.0.2. Fixes [#19362](#).

- [renodr] - Update to libsecret-0.21.4. Fixes [#19358](#).
- [renodr] - Update to glm-1.0.1. Fixes [#19369](#).
- [renodr] - Update to OpenJPEG-2.5.2 (Security Update). Fixes [#19370](#).
- [renodr] - Update to c-ares-1.27.0 (Security Update). Fixes [#19357](#).
- [renodr] - Update to NetworkManager-1.46.0. Fixes [#19350](#).
- [renodr] - Update to Spidermonkey-115.8.0. Fixes [#19344](#).
- [bdubbs] - Update to asciidoctor-2.0.21. Fixes [#19341](#).
- [bdubbs] - Update to mpg123-1.32.5. Fixes [#19328](#).
- [bdubbs] - Update to pytest-8.0.2 (Python module). Fixes [#19326](#).
- [bdubbs] - Update to pcre2-10.43. Fixes [#19062](#).
- [renodr] - Enable support for Vulkan in ffmpeg again. Fixes [#19390](#).
- [renodr] - Update to jasper-4.2.1. Fixes [#19340](#).
- [renodr] - Update to Vulkan-Headers and Vulkan-Loader 1.3.279. Fixes [#19327](#).
- [renodr] - Update to giflib-5.2.2 (Security Update). Fixes [#19335](#).
- [bdubbs] - Update to unrar-7.0.7. Fixes [#18768](#).
- [timtas] - Update to xfce4-panel-4.18.6. Fixes [#19379](#).
- [timtas] - Update to xarchiver-0.5.4.23. Fixes [#19383](#).
- [timtas] - Update to xfce4-terminal-1.1.3. Fixes [#19380](#).
- March 1st, 2024
 - [bdubbs] - Release of BLFS-12.1.

Mailing Lists

The linuxfromscratch.org server is hosting a number of mailing lists that are used for the development of the BLFS book. These lists include, among others, the main development and support lists.

For more information regarding which lists are available, how to subscribe to them, archive locations, etc., visit <https://www.linuxfromscratch.org/mail.html>.

Editor Notes

The BLFS Project has created a Wiki for editors to comment on pages and instructions at <https://wiki.linuxfromscratch.org/blfs/wiki>.

When *editor notes* are present, a link appears in the form <https://wiki.linuxfromscratch.org/blfs/wiki/pkgnane> right below the dependency list. The idea behind the editor notes is to give additional information about the package and/or its build instructions, common pitfalls or maybe even more sophisticated configuration for special cases of use.

The vast majority of the packages do not have editor notes.

Note

The *editor notes* might be outdated. Even though the pages should be reviewed when a package is updated, it might happen that there are notes referring to an obsolete version and therefore, the notes might be out of date. Always check the date of the notes and more importantly, the version of the package the notes refer to.

Asking for Help and the FAQ

If you encounter a problem while using this book, and your problem is not listed in the FAQ (<https://www.linuxfromscratch.org/faq>), you will find that most of the people on Internet Relay Chat (IRC) and on the mailing lists are willing to help you. An overview of the LFS mailing lists can be found in [Mailing lists](#). To assist us in diagnosing and solving your problem, include as much relevant information as possible in your request for help.

Things to Check Prior to Asking

Before asking for help, you should review the following items:

- Is the hardware support compiled into the kernel or available as a module to the kernel? If it is a module, is it configured properly in `modprobe.conf` and has it been loaded? You should use `lsmod` as the `root` user to see if it's loaded. Check the

`sys.log` file or run `modprobe <driver>` to review any error message. If it loads properly, you may need to add the `modprobe` command to your boot scripts.

- Are your permissions properly set, especially for devices? LFS uses groups to make these settings easier, but it also adds the step of adding users to groups to allow access. A simple `usermod -G audio <user>` may be all that's necessary for that user to have access to the sound system. Any question that starts out with "It works as root, but not as ..." requires a thorough review of permissions prior to asking.
- BLFS liberally uses `/opt/<package>`. The main objection to this centers around the need to expand your environment variables for each package placed there (e.g., `PATH=$PATH:/opt/kde/bin`). In most cases, the package instructions will walk you through the changes, but some will not. The section called "[Going Beyond BLFS](#)" is available to help you check.

Things to Mention

Apart from a brief explanation of the problem you're having, the essential things to include in your request are:

- the version of the book you are using (being 12.2),
- the package or section giving you problems,
- the exact error message or symptom you are receiving,
- whether you have deviated from the book or LFS at all,
- if you are installing a BLFS package on a non-LFS system.

(Note that saying that you've deviated from the book doesn't mean that we won't help you. It'll just help us to see other possible causes of your problem.)

Expect guidance instead of specific instructions. If you are instructed to read something, please do so. It generally implies that the answer was way too obvious and that the question would not have been asked if a little research was done prior to asking. The volunteers in the mailing list prefer not to be used as an alternative to doing reasonable research on your end. In addition, the quality of your experience with BLFS is also greatly enhanced by this research, and the quality of volunteers is enhanced because they don't feel that their time has been abused, so they are far more likely to participate.

An excellent article on asking for help on the Internet in general has been written by Eric S. Raymond. It is available online at <http://www.catb.org/~esr/faqs/smarter-questions.html>. Read and follow the hints in that document and you are much more likely to get a response to start with and also to get the help you actually need.

Credits

Many people have contributed both directly and indirectly to BLFS. This page lists all of those we can think of. We may well have left people out and if you feel this is the case, drop us a line. Many thanks to all of the LFS community for their assistance with this project.

Current Editors

- Rahul Chandra
- Bruce Dubbs
- Pierre Labastie
- Ken Moffat
- Douglas Reno
- Xi Ruoyao
- Thomas Trepil

Contributors and Past Editors

The list of contributors is far too large to provide detailed information about the contributions for each contributor. Over the years, the following individuals have provided significant inputs to the book:

- Timothy Bauscher
- Daniel Bauman
- Jeff Bauman
- Andy Benton
- Wayne Blaszczyk
- Paul Campbell

- Nathan Coulson
- Jeroen Coumans
- Guy Dalziel
- Robert Daniels
- Richard Downing
- Manuel Canales Esparcia
- Jim Gifford
- Manfred Glombowski
- Ag Hatzimanikas
- Mark Hymers
- James Iwanek
- David Jensen
- Jeremy Jones
- Seth Klein
- Alex Kloss
- Eric Konopka
- Larry Lawrence
- D-J Lucas
- Chris Lynn
- Andrew McMurry
- Randy McMurchy
- Denis Mugnier
- Billy O'Connor
- Fernando de Oliveira
- Alexander Patrakov
- Olivier Peres
- Andreas Pedersen
- Henning Rohde
- Matt Rogers
- James Robertson
- Henning Rohde
- Chris Staub
- Jesse Tie-Ten-Quee
- Ragnar Thomsen
- Tushar Teredesai
- Jeremy Utley
- Zack Winkles
- Christian Wurst
- Igor Živković

General Acknowledgments

- Fernando Arbeiza
- Miguel Bazdresch

- Gerard Beekmans
- Oliver Brakmann
- Jeremy Byron
- Ian Chilton
- David Ciecielski
- Jim Harris
- Lee Harris
- Marc Heerdink
- Steffen Knollmann
- Eric Konopka
- Scot McPherson
- Ted Riley

Contact Information

Please direct your emails to one of the BLFS mailing lists. See [Mailing lists](#) for more information on the available mailing lists.

Chapter 2. Important Information

This chapter is used to explain some of the policies used throughout the book, to introduce important concepts and to explain some issues you may see with some of the included packages.

Notes on Building Software

Those people who have built an LFS system may be aware of the general principles of downloading and unpacking software. Some of that information is repeated here for those new to building their own software.

Each set of installation instructions contains a URL from which you can download the package. The patches; however, are stored on the LFS servers and are available via HTTP. These are referenced as needed in the installation instructions.

While you can keep the source files anywhere you like, we assume that you have unpacked the package and changed into the directory created by the unpacking process (the source directory). We also assume you have uncompressed any required patches and they are in the directory immediately above the source directory.

We can not emphasize strongly enough that you should start from a *clean source tree* each time. This means that if you have had an error during configuration or compilation, it's usually best to delete the source tree and re-unpack it *before* trying again. This obviously doesn't apply if you're an advanced user used to hacking `Makefile`s and C code, but if in doubt, start from a clean tree.

Building Software as an Unprivileged (non-root) User

The golden rule of Unix System Administration is to use your superpowers only when necessary. Hence, BLFS recommends that you build software as an unprivileged user and only become the `root` user when installing the software. This philosophy is followed in all the packages in this book. Unless otherwise specified, all instructions should be executed as an unprivileged user. The book will advise you on instructions that need `root` privileges.

Unpacking the Software

If a file is in `.tar` format and compressed, it is unpacked by running one of the following commands:

```
tar -xvf filename.tar.gz
tar -xvf filename.tgz
tar -xvf filename.tar.Z
tar -xvf filename.tar.bz2
```

Note

You may omit using the `v` parameter in the commands shown above and below if you wish to suppress the verbose listing of all the files in the archive as they are extracted. This can help speed up the extraction as well as make any errors produced during the extraction more obvious to you.

You can also use a slightly different method:

```
bzcat filename.tar.bz2 | tar -xv
```

Finally, sometimes we have a compressed patch file in `.patch.gz` or `.patch.bz2` format. The best way to apply the patch is piping the output of the decompressor to the `patch` utility. For example:

```
gzip -cd ./patchname.patch.gz | patch -p1
```

Or for a patch compressed with `bzip2`:

```
bzcat ./patchname.patch.bz2 | patch -p1
```

Verifying File Integrity

Generally, to verify that the downloaded file is complete, many package maintainers also distribute md5sums of the files. To verify the md5sum of the downloaded files, download both the file and the corresponding md5sum file to the same directory (preferably from different on-line locations), and (assuming `file.md5sum` is the md5sum file downloaded) run the following command:

```
md5sum -c file.md5sum
```

If there are any errors, they will be reported. Note that the BLFS book includes md5sums for all the source files also. To use the BLFS supplied md5sums, you can create a `file.md5sum` (place the md5sum data and the exact name of the downloaded file on the same line of a file, separated by white space) and run the command shown above. Alternately, simply run the command shown below and compare the output to the md5sum data shown in the BLFS book.

```
md5sum <name_of_downloaded_file>
```

MD5 is not cryptographically secure, so the md5sums are only provided for detecting unmalicious changes to the file content. For example, an error or truncation introduced during network transfer, or a "stealth" update to the package from the upstream (updating the content of a released tarball instead of making a new release properly).

There is no "100%" secure way to make sure the genuity of the source files. Assuming the upstream is managing their website correctly (the private key is not leaked and the domain is not hijacked), and the trust anchors have been set up correctly using [make-ca-1.14](#) on the BLFS system, we can reasonably trust download URLs to the upstream official website **with https protocol**. Note that BLFS book itself is published on a website with https, so you should already have some confidence in https protocol or you wouldn't trust the book content.

If the package is downloaded from an unofficial location (for example a local mirror), checksums generated by cryptographically secure digest algorithms (for example SHA256) can be used to verify the genuity of the package. Download the checksum file from the upstream **official** website (or somewhere **you can trust**) and compare the checksum of the package from unofficial location with it. For example, SHA256 checksum can be checked with the command:

Note

If the checksum and the package are downloaded from the same untrusted location, you won't gain security enhancement by verifying the package with the checksum. The attacker can fake the checksum as well as compromising the package itself.

```
sha256sum -c file.sha256sum
```

If [GnuPG-2.4.5](#) is installed, you can also verify the genuity of the package with a GPG signature. Import the upstream GPG public key with:

```
gpg --recv-key keyID
```

`keyID` should be replaced with the key ID from somewhere **you can trust** (for example, copy it from the upstream official website using https). Now you can verify the signature with:

```
gpg --recv-key file.sig file
```

The advantage of GnuPG signature is, once you imported a public key which can be trusted, you can download both the package and its signature from the same unofficial location and verify them with the public key. So you won't need to

connect to the official upstream website to retrieve a checksum for each new release. You only need to update the public key if it's expired or revoked.

Creating Log Files During Installation

For larger packages, it is convenient to create log files instead of staring at the screen hoping to catch a particular error or warning. Log files are also useful for debugging and keeping records. The following command allows you to create an installation log. Replace `<command>` with the command you intend to execute.

```
( <command> 2>&1 | tee compile.log && exit $PIPESTATUS )
```

`2>&1` redirects error messages to the same location as standard output. The `tee` command allows viewing of the output while logging the results to a file. The parentheses around the command run the entire command in a subshell and finally the `exit $PIPESTATUS` command ensures the result of the `<command>` is returned as the result and not the result of the `tee` command.

Using Multiple Processors

For many modern systems with multiple processors (or cores) the compilation time for a package can be reduced by performing a "parallel make" by either setting an environment variable or telling the make program to simultaneously execute multiple jobs.

For instance, an Intel Core i9-13900K CPU contains 8 performance (P) cores and 16 efficiency (E) cores, and the P cores support SMT (Simultaneous MultiThreading, also known as "Hyper-Threading") so each P core can run two threads simultaneously and the Linux kernel will treat each P core as two logical cores. As a result, there are 32 logical cores in total. To utilize all these logical cores running `make`, we can set an environment variable to tell `make` to run 32 jobs simultaneously:

```
export MAKEFLAGS='-j32'
```

or just building with:

```
make -j32
```

If you have applied the optional `sed` when building ninja in LFS, you can use:

```
export NINJAJOBS=32
```

when a package uses `ninja`, or just:

```
ninja -j32
```

If you are not sure about the number of logical cores, run the `nproc` command.

For `make`, the default number of jobs is 1. But for `ninja`, the default number of jobs is $N + 2$ if the number of logical cores N is greater than 2; or $N + 1$ if N is 1 or 2. The reason to use a number of jobs slightly greater than the number of logical cores is keeping all logical processors busy even if some jobs are performing I/O operations.

Note that the `-j` switches only limits the parallel jobs started by `make` or `ninja`, but each job may still spawn its own processes or threads. For example, `ia.gold` will use multiple threads for linking, and some tests of packages can spawn multiple threads for testing thread safety properties. There is no generic way for the building system to know the number of processes or threads spawned by a job. So generally we should not consider the value passed with `-j` a hard limit of the number of logical cores to use. Read [the section called "Use Linux Control Group to Limit the Resource Usage"](#) if you want to set such a hard limit.

Generally the number of processes should not exceed the number of cores supported by the CPU too much. To list the processors on your system, issue: `grep processor /proc/cpuinfo`.

In some cases, using multiple processes may result in a race condition where the success of the build depends on the order of the commands run by the `make` program. For instance, if an executable needs File A and File B, attempting to link the program before one of the dependent components is available will result in a failure. This condition usually arises because the upstream developer has not properly designated all the prerequisites needed to accomplish a step in the Makefile.

If this occurs, the best way to proceed is to drop back to a single processor build. Adding `-j1` to a `make` command will override the similar setting in the `MAKEFLAGS` environment variable.

Important

Another problem may occur with modern CPU's, which have a lot of cores. Each job started consumes memory, and if the sum of the needed memory for each job exceeds the available memory, you may encounter either an OOM (Out of Memory) kernel interrupt or intense swapping that will slow the build beyond reasonable limits.

Some compilations with `g++` may consume up to 2.5 GB of memory, so to be safe, you should restrict the number of jobs to $(\text{Total Memory in GB})/2.5$, at least for big packages such as LLVM, WebKitGtk, QtWebEngine, or libreoffice.

Use Linux Control Group to Limit the Resource Usage

Sometimes we want to limit the resource usage when we build a package. For example, when we have 8 logical cores, we may want to use only 6 cores for building the package and reserve another 2 cores for playing a movie. The Linux kernel provides a feature called control groups (cgroup) for such a need.

Enable control group in the kernel configuration, then rebuild the kernel and reboot if necessary:

```
General setup --->
[*] Control Group support --->
[*] Memory controller
[*] Cpuset controller
[CGROUPS]
[MEMCG]
[CPUSETS]
```

Ensure [Systemd-256.4](#) and [Shadow-4.16.0](#) have been rebuilt with [Linux-PAM-1.6.1](#) support (if you are interacting via a SSH or graphical session, also ensure the [OpenSSH-9.8p1](#) server or the desktop manager has been built with [Linux-PAM-1.6.1](#)). As the `root` user, create a configuration file to allow resource control without `root` privilege, and instruct `systemd` to reload the configuration:

```
mkdir -pv /etc/systemd/system/user@.service.d &&
cat > /etc/systemd/system/user@.service.d/delegate.conf << EOF &&
[Service]
Delegate=memory cpuset
EOF
systemctl daemon-reload
```

Then logout and login again. Now to run `make -j5` with the first 4 logical cores and 8 GB of system memory, issue:

```
systemctl --user start dbus &&
systemd-run --user --pty --pipe --wait -G -d \
-p MemoryHigh=8G \
-p AllowedCPUs=0-3 \
make -j5
```

With `MemoryHigh=8G`, a soft limit of memory usage is set. If the processes in the cgroup (`make` and all the descendants of it) uses more than 8 GB of system memory in total, the kernel will throttle down the processes and try to reclaim the system memory from them. But they can still use more than 8 GB of system memory. If you want to make a hard limit instead, replace `MemoryHigh` with `MemoryMax`. But doing so will cause the processes killed if 8 GB is not enough for them.

`AllowedCPUs=0-3` makes the kernel only run the processes in the cgroup on the logical cores with numbers 0, 1, 2, or 3. You may need to adjust this setting based the mapping between the logical cores and the physical cores. For example, with an Intel Core i9-13900K CPU, the logical cores 0, 2, 4, ..., 14 are mapped to the first threads of the eight physical P cores, the logical cores 1, 3, 5, ..., 15 are mapped to the second threads of the physical P cores, and the logical cores 16, 17, ..., 31 are mapped to the 16 physical E cores. So if we want to use four threads from four different P cores, we need to specify `0,2,4,6` instead of `0-3`. Note that the other CPU models may use a different mapping scheme. If you are not sure about the mapping between the logical cores and the physical cores, run the `lscpu --extended` command which will output logical core IDs in the `CPU` column, and physical core IDs in the `CORE` column.

When the `nproc` or `ninja` command runs in a cgroup, it will use the number of logical cores assigned to the cgroup as the "system logical core count." For example, in a cgroup with logical cores 0-3 assigned, `nproc` will print 4, and `ninja` will run 6 ($4 + 2$) jobs simultaneously if no `-j` setting is explicitly given.

Read the [man pages `systemd-run\(1\)`](#) and [`systemd.resource-control\(5\)`](#) for the detailed explanation of `parameters` in the command.

Automated Building Procedures

There are times when automating the building of a package can come in handy. Everyone has their own reasons for wanting to automate building, and everyone goes about it in their own way. Creating `Makefiles`, Bash scripts, Perl scripts or simply a list of commands used to cut and paste are just some of the methods you can use to automate building BLFS packages. Detailing how and providing examples of the many ways you can automate the building of packages is beyond the scope of this section. This section will expose you to using file redirection and the `yes` command to help provide ideas on how to automate your builds.

File Redirection to Automate Input

You will find times throughout your BLFS journey when you will come across a package that has a command prompting you for information. This information might be configuration details, a directory path, or a response to a license agreement. This can present a challenge to automate the building of that package. Occasionally, you will be prompted for different

information in a series of questions. One method to automate this type of scenario requires putting the desired responses in a file and using redirection so that the program uses the data in the file as the answers to the questions.

This effectively makes the test suite use the responses in the file as the input to the questions. Occasionally you may end up doing a bit of trial and error determining the exact format of your input file for some things, but once figured out and documented you can use this to automate building the package.

Using `yes` to Automate Input

Sometimes you will only need to provide one response, or provide the same response to many prompts. For these instances, the `yes` command works really well. The `yes` command can be used to provide a response (the same one) to one or more instances of questions. It can be used to simulate pressing just the **Enter** key, entering the **Y** key or entering a string of text. Perhaps the easiest way to show its use is in an example.

First, create a short Bash script by entering the following commands:

```
cat > blfs-yes-test1 << "EOF"
#!/bin/bash

echo -n -e "\n\nPlease type something (or nothing) and press Enter ---> "

read A_STRING

if test "$A_STRING" = ""; then A_STRING="Just the Enter key was pressed"
else A_STRING="You entered '$A_STRING'"
fi

echo -e "\n\n$A_STRING\n\n"
EOF
chmod 755 blfs-yes-test1
```

Now run the script by issuing `./blfs-yes-test1` from the command line. It will wait for a response, which can be anything (or nothing) followed by the **Enter** key. After entering something, the result will be echoed to the screen. Now use the `yes` command to automate the entering of a response:

```
yes | ./blfs-yes-test1
```

Notice that piping `yes` by itself to the script results in **y** being passed to the script. Now try it with a string of text:

```
yes 'This is some text' | ./blfs-yes-test1
```

The exact string was used as the response to the script. Finally, try it using an empty (null) string:

```
yes '' | ./blfs-yes-test1
```

Notice this results in passing just the press of the **Enter** key to the script. This is useful for times when the default answer to the prompt is sufficient. This syntax is used in the [Net-tools](#) instructions to accept all the defaults to the many prompts during the configuration step. You may now remove the test script, if desired.

File Redirection to Automate Output

In order to automate the building of some packages, especially those that require you to read a license agreement one page at a time, requires using a method that avoids having to press a key to display each page. Redirecting the output to a file can be used in these instances to assist with the automation. The previous section on this page touched on creating log files of the build output. The redirection method shown there used the `tee` command to redirect output to a file while also displaying the output to the screen. Here, the output will only be sent to a file.

Again, the easiest way to demonstrate the technique is to show an example. First, issue the command:

```
ls -l /usr/bin | less
```

Of course, you'll be required to view the output one page at a time because the `less` filter was used. Now try the same command, but this time redirect the output to a file. The special file `/dev/null` can be used instead of the filename shown, but you will have no log file to examine:

```
ls -l /usr/bin | less > redirect_test.log 2>&1
```

Notice that this time the command immediately returned to the shell prompt without having to page through the output. You may now remove the log file.

The last example will use the `yes` command in combination with output redirection to bypass having to page through the output and then provide a **y** to a prompt. This technique could be used in instances when otherwise you would have to page

through the output of a file (such as a license agreement) and then answer the question of do you accept the above?. For this example, another short Bash script is required:

```
cat > blfs-yes-test2 << "EOF"
#!/bin/bash

ls -l /usr/bin | less

echo -n -e "\n\nDid you enjoy reading this? (y,n) "

read A_STRING

if test "$A_STRING" = "y"; then A_STRING="You entered the 'y' key"
else A_STRING="You did NOT enter the 'y' key"
fi

echo -e "\n\n$A_STRING\n\n"
EOF
chmod 755 blfs-yes-test2
```

This script can be used to simulate a program that requires you to read a license agreement, then respond appropriately to accept the agreement before the program will install anything. First, run the script without any automation techniques by issuing `./blfs-yes-test2`.

Now issue the following command which uses two automation techniques, making it suitable for use in an automated build script:

```
yes | ./blfs-yes-test2 > blfs-yes-test2.log 2>&1
```

If desired, issue `tail blfs-yes-test2.log` to see the end of the paged output, and confirmation that `y` was passed through to the script. Once satisfied that it works as it should, you may remove the script and log file.

Finally, keep in mind that there are many ways to automate and/or script the build commands. There is not a single "correct" way to do it. Your imagination is the only limit.

Dependencies

For each package described, BLFS lists the known dependencies. These are listed under several headings, whose meaning is as follows:

- *Required* means that the target package cannot be correctly built without the dependency having first been installed, except if the dependency is said to be "runtime" which means the target package can be built but cannot function without it.

Note that a target package can start to "function" in many subtle ways: an installed configuration file can make the init system, cron daemon, or bus daemon to run a program automatically; another package using the target package as a dependency can run a program from the target package in the building system; and the configuration sections in the BLFS book may also run a program from a just installed package. So if you are installing the target package without a *Required (runtime)* dependency installed, You should install the dependency as soon as possible after the installation of the target package.

- *Recommended* means that BLFS strongly suggests this package is installed first (except if said to be "runtime," see below) for a clean and trouble-free build, that won't have issues either during the build process, or at run-time. The instructions in the book assume these packages are installed. Some changes or workarounds may be required if these packages are not installed. If a recommended dependency is said to be "runtime," it means that BLFS strongly suggests that this dependency is installed before using the package, for getting full functionality.
- *Optional* means that this package might be installed for added functionality. Often BLFS will describe the dependency to explain the added functionality that will result. Some optional dependencies are automatically picked up by the target package if the dependency is installed, while others also need additional configuration options to be enabled when the target package is built. Such additional options are often documented in the BLFS book. If an optional dependency is said to be "runtime," it means you may install the dependency after installing the target package to support some optional features of the target package if you need these features.

An optional dependency may be out of BLFS. If you need such an *external* optional dependency for some features you need, read [Going Beyond BLFS](#) for the general hint about installing an out-of-BLFS package.

Using the Most Current Package Sources

On occasion you may run into a situation in the book when a package will not build or work properly. Though the Editors attempt to ensure that every package in the book builds and works properly, sometimes a package has been overlooked or was not tested with this particular version of BLFS.

If you discover that a package will not build or work properly, you should see if there is a more current version of the package. Typically this means you go to the maintainer's web site and download the most current tarball and attempt to build the package. If you cannot determine the maintainer's web site by looking at the download URLs, use Google and

query the package's name. For example, in the Google search bar type: 'package_name download' (omit the quotes) or something similar. Sometimes typing: 'package_name home page' will result in you finding the maintainer's web site.

Stripping One More Time

In LFS, stripping of debugging symbols and unneeded symbol table entries was discussed a couple of times. When building BLFS packages, there are generally no special instructions that discuss stripping again. Stripping can be done while installing a package, or afterwards.

Stripping while Installing a Package

There are several ways to strip executables installed by a package. They depend on the build system used (see below [the section about build systems](#)), so only some generalities can be listed here:

Note

The following methods using the feature of a building system (autotools, meson, or cmake) will not strip static libraries if any is installed. Fortunately there are not too many static libraries in BLFS, and a static library can always be stripped safely by running `strip --strip-unneeded` on it manually.

- The packages using autotools usually have an `install-strip` target in their generated `Makefile` files. So installing stripped executables is just a matter of using `make install-strip` instead of `make install`.
- The packages using the meson build system can accept `-D strip=true` when running `meson`. If you've forgot to add this option running the `meson`, you can also run `meson install --strip` instead of `ninja install`.
- `cmake` generates `install/strip` targets for both the `Unix Makefiles` and `Ninja` generators (the default is `Unix Makefiles` on linux). So just run `make install/strip` OR `ninja install/strip` instead of the `install` counterparts.
- Removing (or not generating) debug symbols can also be achieved by removing the `-g<something>` options in C/C++ calls. How to do that is very specific for each package. And, it does not remove unneeded symbol table entries. So it will not be explained in detail here. See also below the paragraphs about optimization.

Stripping Installed Executables

The `strip` utility changes files in place, which may break anything using it if it is loaded in memory. Note that if a file is in use but just removed from the disk (i.e. not overwritten nor modified), this is not a problem since the kernel can use "deleted" files. Look at `/proc/*/*maps` and it is likely that you'll see some (*deleted*) entries. The `mv` just removes the destination file from the directory but does not touch its content, so that it satisfies the condition for the kernel to use the old (deleted) file. But this approach can detach hard links into duplicated copies, causing a bloat which is obviously unwanted as we are stripping to reduce system size. If two files in a same file system share the same inode number, they are hard links to each other and we should reconstruct the link. The script below is just an example. It should be run as the `root` user:

```
cat > /usr/sbin/strip-all.sh << "EOF"
#!/usr/bin/bash

if [ $EUID -ne 0 ]; then
    echo "Need to be root"
    exit 1
fi

last_fs_inode=
last_file=

{ find /usr/lib -type f -name '*.so*' ! -name '*dbg'
  find /usr/lib -type f -name '*.a'
  find /usr/{bin,sbin,libexec} -type f
} | xargs stat -c '%m %i %n' | sort | while read fs inode file; do
    if ! readelf -h $file >/dev/null 2>&1; then continue; fi
    if file $file | grep --quiet --invert-match 'not stripped'; then continue; fi

    if [ "$fs $inode" = "$last_fs_inode" ]; then
        ln -f $last_file $file;
        continue;
    fi

    cp --preserve $file ${file}.tmp
    strip --strip-unneeded ${file}.tmp
    mv ${file}.tmp $file

    last_fs_inode="$fs $inode"
```

```

last_file=$file
done
EOF
chmod 744 /usr/sbin(strip-all.sh)

```

If you install programs in other directories such as `/opt` or `/usr/local`, you may want to strip the files there too. Just add other directories to scan in the compound list of `find` commands between the braces.

For more information on stripping, see <https://www.technovelty.org/linux/stripping-shared-libraries.html>.

Working with different build systems

There are now three different build systems in common use for converting C or C++ source code into compiled programs or libraries and their details (particularly, finding out about available options and their default values) differ. It may be easiest to understand the issues caused by some choices (typically slow execution or unexpected use of, or omission of, optimizations) by starting with the `CFLAGS`, `CXXFLAGS`, and `LDFLAGS` environment variables. There are also some programs which use Rust.

Most LFS and BLFS builders are probably aware of the basics of `CFLAGS` and `CXXFLAGS` for altering how a program is compiled. Typically, some form of optimization is used by upstream developers (`-O2` or `-O3`), sometimes with the creation of debug symbols (`-g`), as defaults.

If there are contradictory flags (e.g. multiple different `-O` values), the *last* value will be used. Sometimes this means that flags specified in environment variables will be picked up before values hardcoded in the Makefile, and therefore ignored. For example, where a user specifies `-O2` and that is followed by `-O3` the build will use `-O3`.

There are various other things which can be passed in `CFLAGS` or `CXXFLAGS`, such as allowing using the instruction set extensions available with a specific microarchitecture (e.g. `-march=amdfam10` or `-march=native`), tune the generated code for a specific microarchitecture (e.g. `-mtune=tigerlake` or `-mtune=native`, if `-mtune=` is not used, the microarchitecture from `-march=` setting will be used), or specifying a specific standard for C or C++ (`-std=c++17` for example). But one thing which has now come to light is that programmers might include debug assertions in their code, expecting them to be disabled in releases by using `-D NDEBUG`. Specifically, if [Mesa-24.1.5](#) is built with these assertions enabled, some activities such as loading levels of games can take extremely long times, even on high-class video cards.

Autotools with Make

This combination is often described as “CMMI” (configure, make, make install) and is used here to also cover the few packages which have a configure script that is not generated by autotools.

Sometimes running `./configure --help` will produce useful options about switches which might be used. At other times, after looking at the output from configure you may need to look at the details of the script to find out what it was actually searching for.

Many configure scripts will pick up any `CFLAGS` or `CXXFLAGS` from the environment, but CMMI packages vary about how these will be mixed with any flags which would otherwise be used (*variously*: ignored, used to replace the programmer's suggestion, used before the programmer's suggestion, or used after the programmer's suggestion).

In most CMMI packages, running `make` will list each command and run it, interspersed with any warnings. But some packages try to be “silent” and only show which file they are compiling or linking instead of showing the command line. If you need to inspect the command, either because of an error, or just to see what options and flags are being used, adding `v=1` to the `make` invocation may help.

CMake

CMake works in a very different way, and it has two backends which can be used on BLFS: `make` and `ninja`. The default backend is `make`, but `ninja` can be faster on large packages with multiple processors. To use `ninja`, specify `-G Ninja` in the `cmake` command. However, there are some packages which create fatal errors in their `ninja` files but build successfully using the default of Unix Makefiles.

The hardest part of using CMake is knowing what options you might wish to specify. The only way to get a list of what the package knows about is to run `cmake -LAH` and look at the output for that default configuration.

Perhaps the most-important thing about CMake is that it has a variety of `CMAKE_BUILD_TYPE` values, and these affect the flags. The default is that this is not set and no flags are generated. Any `CFLAGS` or `CXXFLAGS` in the environment will be used. If the programmer has coded any debug assertions, those will be enabled unless `-D NDEBUG` is used. The following `CMAKE_BUILD_TYPE` values will generate the flags shown, and these will come after any flags in the environment and therefore take precedence.

Value	Flags
Debug	<code>-g</code>
Release	<code>-O3 -D NDEBUG</code>
RelWithDebInfo	<code>-O2 -g -D NDEBUG</code>

Value	Flags
MinSizeRel	-Os -D NDEBUG

CMake tries to produce quiet builds. To see the details of the commands which are being run, use `make VERBOSE=1` or `ninja -v`.

By default, CMake treats file installation differently from the other build systems: if a file already exists and is not newer than a file that would overwrite it, then the file is not installed. This may be a problem if a user wants to record which file belongs to a package, either using `LD_PRELOAD`, or by listing files newer than a timestamp. The default can be changed by setting the variable `CMAKE_INSTALL_ALWAYS` to 1 in the environment, for example by `export`'ing it.

Meson

Meson has some similarities to CMake, but many differences. To get details of the defines that you may wish to change you can look at `meson_options.txt` which is usually in the top-level directory.

If you have already configured the package by running `meson` and now wish to change one or more settings, you can either remove the build directory, recreate it, and use the altered options, or within the build directory run `meson configure`, e.g. to set an option:

```
meson configure -D <some_option>=true
```

If you do that, the file `meson-private/cmd_line.txt` will show the *last* commands which were used.

Meson provides the following buildtype values, and the flags they enable come *after* any flags supplied in the environment and therefore take precedence.

- plain: no added flags. This is for distributors to supply their own `CFLAGS`, `CXXFLAGS` and `LDFLAGS`. There is no obvious reason to use this in BLFS.
- debug: `-g` - this is the default if nothing is specified in either `meson.build` or the command line. However it results large and slow binaries, so we should override it in BLFS.
- debugoptimized: `-O2 -g` - this is the default specified in `meson.build` of some packages.
- release: `-O3` (occasionally a package will force `-O2` here) - this is the buildtype we use for most packages with Meson build system in BLFS.

The `-D NDEBUG` flag is implied by the release buildtype for some packages (for example [Mesa-24.1.5](#)). It can also be provided explicitly by passing `-D b_nndebug=true`.

To see the details of the commands which are being run in a package using `meson`, use `ninja -v`.

Rustc and Cargo

Most released rustc programs are provided as crates (source tarballs) which will query a server to check current versions of dependencies and then download them as necessary. These packages are built using `cargo --release`. In theory, you can manipulate the `RUSTFLAGS` to change the optimize-level (default for `--release` is 3, i. e. `-Copt-level=3`, like `-O3`) or to force it to build for the machine it is being compiled on, using `-Ctarget-cpu=native` but in practice this seems to make no significant difference.

If you are compiling a standalone Rust program (as an unpackaged `.rs` file) by running `rustc` directly, you should specify `-O` (the abbreviation of `-Copt-level=2`) or `-Copt-level=3` otherwise it will do an unoptimized compile and run *much* slower. If you are compiling the program for debugging it, replace the `-O` or `-Copt-level=` options with `-g` to produce an unoptimized program with debug info.

Like `ninja`, by default `cargo` uses all logical cores. This can often be worked around, either by exporting `CARGO_BUILD_JOBS=<N>` or passing `--jobs <N>` to `cargo`. For compiling rustc itself, specifying `--jobs <N>` for invocations of `*.py` (together with the `CARGO_BUILD_JOBS` environment variable, which looks like a "belt and braces" approach but seems to be necessary) mostly works. The exception is running the tests when building rustc, some of them will nevertheless use all online CPUs, at least as of rustc-1.42.0.

Optimizing the build

Many people will prefer to optimize compiles as they see fit, by providing `CFLAGS` or `CXXFLAGS`. For an introduction to the options available with `gcc` and `g++` see <https://gcc.gnu.org/onlinedocs/gcc-14.2.0/gcc/Optimize-Options.html>. The same content can be also found in `info gcc`.

Some packages default to `-O2 -g`, others to `-O3 -g`, and if `CFLAGS` or `CXXFLAGS` are supplied they might be added to the package's defaults, replace the package's defaults, or even be ignored. There are details on some desktop packages which were mostly current in April 2019 at <https://www.linuxfromscratch.org/~ken/tuning/> - in particular, `README.txt`, `tuning-1-packages-and-notes.txt`, and `tuning-notes-2B.txt`. The particular thing to remember is that if you want to try some of the more interesting flags you may need to force verbose builds to confirm what is being used.

Clearly, if you are optimizing your own program you can spend time to profile it and perhaps recode some of it if it is too slow. But for building a whole system that approach is impractical. In general, `-O3` usually produces faster programs than `-O2`. Specifying `-march=native` is also beneficial, but means that you cannot move the binaries to an incompatible machine - this can also apply to newer machines, not just to older machines. For example programs compiled for `amdfam10` run on old Phenoms, Kaveris, and Ryzens, but programs compiled for a Kaveri will not run on a Ryzen because certain op-codes are not present. Similarly, if you build for a Haswell not everything will run on a SandyBridge.

Note

Be careful that the name of a `-march` setting does not always match the baseline of the microarchitecture with the same name. For example, the Skylake-based Intel Celeron processors do not support AVX at all, but `-march=skylake` assumes AVX and even AVX2.

When a shared library is built by GCC, a feature named "semantic interposition" is enabled by default. When the shared library refers to a symbol name with external linkage and default visibility, if the symbol exists in both the shared library and the main executable, semantic interposition guarantees the symbol in the main executable is always used. This feature was invented in an attempt to make the behavior of linking a shared library and linking a static library as similar as possible. Today only a small number of packages still depend on semantic interposition, but the feature is still on by the default of GCC, causing many optimizations disabled for shared libraries because they conflict with semantic interposition. The `-fno-semantic-interposition` option can be passed to `gcc` or `g++` to disable semantic interposition and enable more optimizations for shared libraries. This option is used as the default of some packages (for example [Python-3.12.5](#)), and it's also the default of Clang.

There are also various other options which some people claim are beneficial. At worst, you get to recompile and test, and then discover that in your usage the options do not provide a benefit.

If building Perl or Python modules, in general the `CFLAGS` and `CXXFLAGS` used are those which were used by those "parent" packages.

For `LDFLAGS`, there are three options can be used for optimization. They are quite safe to use and the building system of some packages use some of these options as the default.

With `-Wl,-O1`, the linker will optimize the hash table to speed up the dynamic linking. Note that `-Wl,-O1` is completely unrelated to the compiler optimization flag `-O1`.

With `-Wl,--as-needed`, the linker will disregard unnecessary `-lfoo` options from the command line, i. e. the shared library `libfoo` will only be linked if a symbol in `libfoo` is really referred from the executable or shared library being linked. This can sometimes mitigate the "excessive dependencies to shared libraries" issues caused by libtool.

With `-Wl,-z,pack-relative-relocs`, the linker generates a more compacted form of the relative relocation entries for PIEs and shared libraries. It reduces the size of the linked PIE or shared library, and speeds up the loading of the PIE or shared library.

The `-Wl,` prefix is necessary because despite the variable is named `LDFLAGS`, its content is actually passed to `gcc` (or `g++`, `clang`, etc.) during the link stage, not directly passed to `ld`.

Options for hardening the build

Even on desktop systems, there are still a lot of exploitable vulnerabilities. For many of these, the attack comes via javascript in a browser. Often, a series of vulnerabilities are used to gain access to data (or sometimes to pwn, i.e. own, the machine and install rootkits). Most commercial distros will apply various hardening measures.

In the past, there was Hardened LFS where `gcc` (a much older version) was forced to use hardening (with options to turn some of it off on a per-package basis). The current LFS and BLFS books are carrying forward a part of its spirit by enabling PIE (`-fPIE -pie`) and SSP (`-fstack-protector-strong`) as the defaults for `GCC` and `clang`. And, the linkers (`ld.bfd` and `ld.gold`) have also enabled `-Wl,-z,relro` which makes a part of the Global Offset Table (GOT) immutable, by default since Binutils 2.27. What is being covered here is different - first you have to make sure that the package is indeed using your added flags and not over-riding them.

For hardening options which are reasonably cheap, there is some discussion in the 'tuning' link above (occasionally, one or more of these options might be inappropriate for a package). These options are `-D _FORTIFY_SOURCE=2` (or `-D _FORTIFY_SOURCE=3` which is more secure but with a larger performance overhead) and (for C++) `-D _GLIBCXX_ASSERTIONS`. On modern machines these should only have a little impact on how fast things run, and often they will not be noticeable.

The main distros use much more, such as:

- `-Wl,-z,now`: disables lazy binding to enhance `-Wl,-z,relro`, so the *entire* GOT can be made immutable.
- `-fstack-clash-protection`: prevents the attacker from using an offset large enough and not adequately checked to jump over the stack guard page placed by the kernel and the stack canary placed by `-fstack-protector=strong`, and modify the stack from a heap address, or vice versa.

- `-ftrivial-auto-var-init=zero`: initializes some variables by filling zero bytes if they are not initialized by other means.
- `-fcf-protection=full`: utilizes Intel and AMD CET technology to limit the target addresses of control-flow transfer instructions. To make it really effective for a package, all packages providing a shared library for the package to use must be built with this option, as well as that package itself, Glibc must be configured with the `--enable-cet` option enabled, and the system must run on Intel Tiger Lake or newer, or AMD Zen 3 or newer. If the criteria is not met the program compiled with this option will still run, but not really protected by CET.

In GCC 14, the option `-fhardened` is a shorthand to enable all the hardening options mentioned above. It sets `-D_FORTIFY_SOURCE=3` instead of `-D_FORTIFY_SOURCE=2`.

You may also encounter the so-called “userspace retpoline” (`-mindirect-branch=thunk` etc.) which is the equivalent of the spectre mitigations applied to the linux kernel in late 2018. The kernel mitigations caused a lot of complaints about lost performance, if you have a production server you might wish to consider testing that, along with the other available options, to see if performance is still sufficient.

Whilst gcc has many hardening options, clang/LLVM's strengths lie elsewhere. Some options which gcc provides are said to be less effective in clang/LLVM.

The /usr Versus /usr/local Debate

Should I install XXX in /usr or /usr/local?

This is a question without an obvious answer for an LFS based system.

In traditional Unix systems, `/usr` usually contains files that come with the system distribution, and the `/usr/local` tree is free for the local administrator to manage. The only really hard and fast rule is that Unix distributions should not touch `/usr/local`, except perhaps to create the basic directories within it.

With Linux distributions like Red Hat, Debian, etc., a possible rule is that `/usr` is managed by the distribution's package system and `/usr/local` is not. This way the package manager's database knows about every file within `/usr`.

LFS users build their own system and so deciding where the system ends and local files begin is not straightforward. So the choice should be made in order to make things easier to administer. There are several reasons for dividing files between `/usr` and `/usr/local`.

- On a network of several machines all running LFS, or mixed LFS and other Linux distributions, `/usr/local` could be used to hold packages that are common between all the computers in the network. It can be NFS mounted or mirrored from a single server. Here local indicates local to the site.
- On a network of several computers all running an identical LFS system, `/usr/local` could hold packages that are different between the machines. In this case local refers to the individual computers.
- Even on a single computer, `/usr/local` can be useful if you have several distributions installed simultaneously, and want a place to put packages that will be the same on all of them.
- Or you might regularly rebuild your LFS, but want a place to put files that you don't want to rebuild each time. This way you can wipe the LFS file system and start from a clean partition every time without losing everything.

Some people ask why not use your own directory tree, e.g., `/usr/site`, rather than `/usr/local`?

There is nothing stopping you, many sites do make their own trees, however it makes installing new software more difficult. Automatic installers often look for dependencies in `/usr` and `/usr/local`, and if the file it is looking for is in `/usr/site` instead, the installer will probably fail unless you specifically tell it where to look.

What is the BLFS position on this?

All of the BLFS instructions install programs in `/usr` with optional instructions to install into `/opt` for some specific packages.

Optional Patches

As you follow the various sections in the book, you will observe that the book occasionally includes patches that are required for a successful and secure installation of the packages. The general policy of the book is to include patches that fall in one of the following criteria:

- Fixes a compilation problem.
- Fixes a security problem.
- Fixes a broken functionality.

In short, the book only includes patches that are either required or recommended. There is a [Patches subproject](#) which hosts various patches (including the patches referenced in the books) to enable you to configure your LFS the way you like

it.

BLFS Systemd Units

The BLFS Systemd Units package contains the systemd unit files that are used throughout the book.

Package Information

- Download: <https://www.linuxfromscratch.org/blfs/downloads/12.2-systemd/blfs-systemd-units-20240801.tar.xz>

The BLFS Systemd Units package will be used throughout the BLFS book for systemd unit files. Each systemd unit has a separate install target. It is recommended that you keep the package source directory around until completion of your BLFS system. When a systemd unit is requested from BLFS Systemd Units, simply change to the directory, and as the *root* user, execute the given `make install-<systemd-unit>` command. This command installs the systemd unit to its proper location (along with any auxiliary configuration scripts) and also enables it by default.

Note

It is advisable to peruse each systemd unit before installation to determine whether the installed files meet your needs.

About Libtool Archive (.la) files

Files with a .la extension

In LFS and BLFS, many packages use an internally shipped libtool copy to build on a variety of Unix platforms. This includes platforms such as AIX, Solaris, IRIX, HP-UX, and Cygwin as well as Linux. The origins of this tool are quite dated. It was intended to manage libraries on systems with less advanced capabilities than a modern Linux system.

On a Linux system, libtool specific files are generally unneeded. Normally libraries are specified in the build process during the link phase. Since a linux system uses the [Executable and Linkable Format \(ELF\)](#), for executables and dynamic libraries, information needed to complete the task is embedded in the files. Both the linker and the program loader can query the appropriate files and properly link or execute the program.

Static libraries are rarely used in LFS and BLFS. And, nowadays most packages store the information needed for linking against a static library into a .pc file, instead of relying on libtool. A `pkg-config --static --libs` command will output the sufficient flags for the linker to link against a static library without any libtool magic.

The problem is that libtool usually creates one or more text files for package libraries called libtool archives. These small files have a ".la" extension and contain information that is similar to that embedded in the libraries or pkg-config files. When building a package that uses libtool, the process automatically looks for these files. Sometimes a .la file can contains the name or path of a static library used during build but not installed, then the build process will break because the .la file refers to something nonexistent on the system. Similarly, if a package is updated and no longer uses the .la file, then the build process can break with the old .la files.

The solution is to remove the .la files. However there is a catch. Some packages, such as [ImageMagick-7.1.1-36](#), use a libtool function, `lt_dlopen`, to load libraries as needed during execution and resolve their dependencies at run time. In this case, the .la files should remain.

The script below, removes all unneeded .la files and saves them in a directory, /var/local/la-files by default, not in the normal library path. It also searches all pkg-config files (.pc) for embedded references to .la files and fixes them to be conventional library references needed when an application or library is built. It can be run as needed to clean up the directories that may be causing problems.

```
cat > /usr/sbin/remove-la-files.sh << "EOF"
#!/bin/bash

# /usr/sbin/remove-la-files.sh
# Written for Beyond Linux From Scratch
# by Bruce Dubbs <bdubbs@linuxfromscratch.org>

# Make sure we are running with root privs
if test "${EUID}" -ne 0; then
    echo "Error: $(basename ${0}) must be run as the root user! Exiting..."
    exit 1
fi

# Make sure PKG_CONFIG_PATH is set if discarded by sudo
source /etc/profile
```

```

OLD_LA_DIR=/var/local/la-files

mkdir -p $OLD_LA_DIR

# Only search directories in /opt, but not symlinks to directories
OPTDIRS=$(find /opt -mindepth 1 -maxdepth 1 -type d)

# Move any found .la files to a directory out of the way
find /usr/lib $OPTDIRS -name "*.la" ! -path "/usr/lib/ImageMagick*" \
-exec mv -fv {} $OLD_LA_DIR \;
#####
#####

# Fix any .pc files that may have .la references

STD_PC_PATH='/usr/lib/pkgconfig
/usr/share/pkgconfig
/usr/local/lib/pkgconfig
/usr/local/share/pkgconfig'

# For each directory that can have .pc files
for d in $(echo $PKG_CONFIG_PATH | tr : ' ') $STD_PC_PATH; do

    # For each pc file
    for pc in $d/*.pc ; do
        if [ $pc == "$d/*.pc" ]; then continue; fi

        # Check each word in a line with a .la reference
        for word in $(grep '\.la' $pc); do
            if $(echo $word | grep -q '\.la$'); then
                mkdir -p $d/la-backup
                cp -fv $pc $d/la-backup

                basename=$(basename $word)
                libref=$(echo $basename|sed -e 's/^lib/-l/' -e 's/\.la//')

                # Fix the .pc file
                sed -i "s:$word:$libref:" $pc
            fi
        done
    done
done

EOF

chmod +x /usr/sbin/remove-la-files.sh

```

Libraries: Static or shared?

Libraries: Static or shared?

The original libraries were simply an archive of routines from which the required routines were extracted and linked into the executable program. These are described as static libraries, with names of the form `libfoo.a` on UNIX-like operating systems. On some old operating systems they are the only type available.

On almost all Linux platforms there are also “shared” (or equivalently “dynamic”) libraries (with names of the form `libfoo.so`) – one copy of the library is loaded into virtual memory, and shared by all the programs which call any of its functions. This is space efficient.

In the past, essential programs such as a shell were often linked statically so that some form of minimal recovery system would exist even if shared libraries, such as `libc.so`, became damaged (e.g. moved to `lost+found` after `fsck` following an unclean shutdown). Nowadays, most people use an alternative system install or a USB stick if they have to recover. Journaling filesystems also reduce the likelihood of this sort of problem.

Within the book, there are various places where configure switches such as `--disable-static` are employed, and other places where the possibility of using system versions of libraries instead of the versions included within another package is discussed. The main reason for this is to simplify updates of libraries.

If a package is linked to a dynamic library, updating to a newer library version is automatic once the newer library is installed and the program is (re)started (provided the library major version is unchanged, e.g. going from `libfoo.so.2.0` to `libfoo.so.2.1`. Going to `libfoo.so.3` will require recompilation – `ldd` can be used to find which programs use the old version). If a program is linked to a static library, the program always has to be recompiled. If you know which programs are linked to a particular static library, this is merely an annoyance. But usually you will *not* know which programs to recompile.

One way to identify when a static library is used, is to deal with it at the end of the installation of every package. Write a script to find all the static libraries in `/usr/lib` or wherever you are installing to, and either move them to another directory so that they are no longer found by the linker, or rename them so that `libfoo.a` becomes e.g. `libfoo.a.hidden`. The static library can then be temporarily restored if it is ever needed, and the package needing it can be identified. This shouldn't be done blindly since many libraries only exist in a static version. For example, some libraries from the glibc and gcc packages should always be present on the system (`libc_nonshared.a`, `libg.a`, `libpthread_nonshared.a`, `libssp_nonshared.a`, `libsopc++.a` as of glibc-2.36 and gcc-12.2).

If you use this approach, you may discover that more packages than you were expecting use a static library. That was the case with nettle-2.4 in its default static-only configuration: It was required by GnuTLS-3.0.19, but also linked into package(s) which used GnuTLS, such as glib-networking-2.32.3.

Many packages put some of their common functions into a static library which is only used by the programs within the package and, crucially, the library is *not* installed as a standalone library. These internal libraries are not a problem – if the package has to be rebuilt to fix a bug or vulnerability, nothing else is linked to them.

When BLFS mentions system libraries, it means shared versions of libraries. Some packages such as [Firefox-128.1.0](#) and [ghostscript-10.03.1](#) bundle many other libraries in their build tree. The version they ship is often older than the version used in the system, so it may contain bugs – sometimes developers go to the trouble of fixing bugs in their included libraries, other times they do not.

Sometimes, deciding to use system libraries is an easy decision. Other times it may require you to alter the system version (e.g. for [libpng-1.6.43](#) if used for [Firefox-128.1.0](#)). Occasionally, a package ships an old library and can no longer link to the current version, but can link to an older version. In this case, BLFS will usually just use the shipped version. Sometimes the included library is no longer developed separately, or its upstream is now the same as the package's upstream and you have no other packages which will use it. In those cases, you'll be lead to use the included library even if you usually prefer to use system libraries.

Locale Related Issues

This page contains information about locale related problems and issues. In the following paragraphs you'll find a generic overview of things that can come up when configuring your system for various locales. Many (but not all) existing locale related problems can be classified and fall under one of the headings below. The severity ratings below use the following criteria:

- Critical: The program doesn't perform its main function. The fix would be very intrusive, it's better to search for a replacement.
- High: Part of the functionality that the program provides is not usable. If that functionality is required, it's better to search for a replacement.
- Low: The program works in all typical use cases, but lacks some functionality normally provided by its equivalents.

If there is a known workaround for a specific package, it will appear on that package's page.

The Needed Encoding is Not a Valid Option in the Program

Severity: Critical

Some programs require the user to specify the character encoding for their input or output data and present only a limited choice of encodings. This is the case for the `-x` option in [Enscript-1.6.6](#), the `-input-charset` option in unpatched [Cdrtools-3.02a09](#), and the character sets offered for display in the menu of [Links-2.30](#). If the required encoding is not in the list, the program usually becomes completely unusable. For non-interactive programs, it may be possible to work around this by converting the document to a supported input character set before submitting to the program.

A solution to this type of problem is to implement the necessary support for the missing encoding as a patch to the original program or to find a replacement.

The Program Assumes the Locale-Based Encoding of External Documents

Severity: High for non-text documents, low for text documents

Some programs, [nano-8.1](#) or [JOE-4.6](#) for example, assume that documents are always in the encoding implied by the current locale. While this assumption may be valid for the user-created documents, it is not safe for external ones. When this assumption fails, non-ASCII characters are displayed incorrectly, and the document may become unreadable.

If the external document is entirely text based, it can be converted to the current locale encoding using the `iconv` program.

For documents that are not text-based, this is not possible. In fact, the assumption made in the program may be completely invalid for documents where the Microsoft Windows operating system has set de facto standards. An example of this problem is ID3v1 tags in MP3 files. For these cases, the only solution is to find a replacement program that doesn't have the issue (e.g., one that will allow you to specify the assumed document encoding).

Among BLFS packages, this problem applies to [nano-8.1](#), [JOE-4.6](#), and all media players except [Audacious-4.4](#).

Another problem in this category is when someone cannot read the documents you've sent them because their operating system is set up to handle character encodings differently. This can happen often when the other person is using Microsoft Windows, which only provides one character encoding for a given country. For example, this causes problems with UTF-8 encoded TeX documents created in Linux. On Windows, most applications will assume that these documents have been created using the default Windows 8-bit encoding.

In extreme cases, Windows encoding compatibility issues may be solved only by running Windows programs under [Wine](#).

The Program Uses or Creates Filenames in the Wrong Encoding

Severity: Critical

The POSIX standard mandates that the filename encoding is the encoding implied by the current LC_CTYPE locale category. This information is well-hidden on the page which specifies the behavior of Tar and Cpio programs. Some programs get it wrong by default (or simply don't have enough information to get it right). The result is that they create filenames which are not subsequently shown correctly by `ls`, or they refuse to accept filenames that `ls` shows properly. For the [GLib-2.80.4](#) library, the problem can be corrected by setting the `G_FILENAME_ENCODING` environment variable to the special "@locale" value. Glib2 based programs that don't respect that environment variable are buggy.

The [Zip-3.0](#) and [UnZip-6.0](#) have this problem because they hard-code the expected filename encoding. UnZip contains a hard-coded conversion table between the CP850 (DOS) and ISO-8859-1 (UNIX) encodings and uses this table when extracting archives created under DOS or Microsoft Windows. However, this assumption only works for those in the US and not for anyone using a UTF-8 locale. Non-ASCII characters will be mangled in the extracted filenames.

The general rule for avoiding this class of problems is to avoid installing broken programs. If this is impossible, the [convmv](#) command-line tool can be used to fix filenames created by these broken programs, or intentionally mangle the existing filenames to meet the broken expectations of such programs.

In other cases, a similar problem is caused by importing filenames from a system using a different locale with a tool that is not locale-aware (e.g., [OpenSSH-9.8p1](#)). In order to avoid mangling non-ASCII characters when transferring files to a system with a different locale, any of the following methods can be used:

- Transfer anyway, fix the damage with `convmv`.
- On the sending side, create a tar archive with the `--format posix` switch passed to `tar` (this will be the default in a future version of `tar`).
- Mail the files as attachments. Mail clients specify the encoding of attached filenames.
- Write the files to a removable disk formatted with a FAT or FAT32 filesystem.
- Transfer the files using Samba.
- Transfer the files via FTP using RFC2640-aware server (this currently means only wu-ftpd, which has bad security history) and client (e.g., lftp).

The last four methods work because the filenames are automatically converted from the sender's locale to UNICODE and stored or sent in this form. They are then transparently converted from UNICODE to the recipient's locale encoding.

The Program Breaks Multibyte Characters or Doesn't Count Character Cells Correctly

Severity: High or critical

Many programs were written in an older era where multibyte locales were not common. Such programs assume that C "char" data type, which is one byte, can be used to store single characters. Further, they assume that any sequence of characters is a valid string and that every character occupies a single character cell. Such assumptions completely break in UTF-8 locales. The visible manifestation is that the program truncates strings prematurely (i.e., at 80 bytes instead of 80 characters). Terminal-based programs don't place the cursor correctly on the screen, don't react to the "Backspace" key by erasing one character, and leave junk characters around when updating the screen, usually turning the screen into a complete mess.

Fixing this kind of problems is a tedious task from a programmer's point of view, like all other cases of retrofitting new concepts into the old flawed design. In this case, one has to redesign all data structures in order to accommodate to the fact that a complete character may span a variable number of "char"s (or switch to `wchar_t` and convert as needed). Also, for every call to the "strlen" and similar functions, find out whether a number of bytes, a number of characters, or the width of the string was really meant. Sometimes it is faster to write a program with the same functionality from scratch.

Among BLFS packages, this problem applies to [xine-ui-0.99.14](#) and all the shells.

The Package Installs Manual Pages in Incorrect or Non-Displayable Encoding

Severity: Low

LFS expects that manual pages are in the language-specific (usually 8-bit) encoding, as specified on the [LFS Man DB page](#). However, some packages install translated manual pages in UTF-8 encoding (e.g., Shadow, already dealt with), or manual pages in languages not in the table. Not all BLFS packages have been audited for conformance with the requirements put in LFS (the large majority have been checked, and fixes placed in the book for packages known to install non-conforming

manual pages). If you find a manual page installed by any of BLFS packages that is obviously in the wrong encoding, please remove or convert it as needed, and report this to BLFS team as a bug.

You can easily check your system for any non-conforming manual pages by copying the following short shell script to some accessible location,

```
#!/bin/sh
# Begin checkman.sh
# Usage: find /usr/share/man -type f | xargs checkman.sh
for a in "$@"
do
    # echo "Checking $a..."
    # Pure-ASCII manual page (possibly except comments) is OK
    grep -v '\\"' "$a" | iconv -f US-ASCII -t US-ASCII >/dev/null 2>&1 \
        && continue
    # Non-UTF-8 manual page is OK
    iconv -f UTF-8 -t UTF-8 "$a" >/dev/null 2>&1 || continue
    # Found a UTF-8 manual page, bad.
    echo "UTF-8 manual page: $a" >&2
done
# End checkman.sh
```

and then issuing the following command (modify the command below if the `checkman.sh` script is not in your `PATH` environment variable):

```
find /usr/share/man -type f | xargs checkman.sh
```

Note that if you have manual pages installed in any location other than `/usr/share/man` (e.g., `/usr/local/share/man`), you must modify the above command to include this additional location.

Going Beyond BLFS

The packages that are installed in this book are only the tip of the iceberg. We hope that the experience you gained with the LFS book and the BLFS book will give you the background needed to compile, install and configure packages that are not included in this book.

When you want to install a package to a location other than `/`, or `/usr`, you are installing outside the default environment settings on most machines. The following examples should assist you in determining how to correct this situation. The examples cover the complete range of settings that may need updating, but they are not all needed in every situation.

- Expand the `PATH` to include `$PREFIX/bin`.
- Expand the `PATH` for `root` to include `$PREFIX/sbin`.
- Add `$PREFIX/lib` to `/etc/ld.so.conf` or expand `LD_LIBRARY_PATH` to include it. Before using the latter option, check out http://xahlee.info/UnixResource_dir/_ldpath.html. If you modify `/etc/ld.so.conf`, remember to update `/etc/ld.so.cache` by executing `ldconfig` as the `root` user.
- Add `$PREFIX/man` to `/etc/man_db.conf`.
- Add `$PREFIX/info` to `INFOPATH`.
- Add `$PREFIX/lib/pkgconfig` to `PKG_CONFIG_PATH`. Some packages are now installing `.pc` files in `$PREFIX/share/pkgconfig`, so you may have to include this directory also.
- Add `$PREFIX/include` to `CPPFLAGS` when compiling packages that depend on the package you installed.
- Add `$PREFIX/lib` to `LDLIBRARY_PATH` when compiling packages that depend on a library installed by the package.

If you are in search of a package that is not in the book, the following are different ways you can search for the desired package.

- If you know the name of the package, then search SourceForge for it at <https://sourceforge.net/directory/>, and search GitHub for it at <https://github.com/>. Also search Google at <https://google.com/>. Sometimes a search for the `rpm` at <https://rpmfind.net/> or the `deb` at https://www.debian.org/distrib/packages#search_packages can also lead to a link to the package.
- If you know the name of the executable, but not the package that the executable belongs to, first try a Google search with the name of the executable. If the results are overwhelming, try searching for the given executable in the Debian repository at https://www.debian.org/distrib/packages#search_contents.

Some general hints on handling new packages:

- Many of the newer packages follow the `./configure && make && make install` process. Help on the options accepted by `configure` can be obtained via the command `./configure --help`.
- Most of the packages contain documentation on compiling and installing the package. Some of the documents are excellent, some not so excellent. Check out the homepage of the package for any additional and updated hints for compiling and configuring the package.
- If you are having a problem compiling the package, try searching the LFS archives at <https://www.linuxfromscratch.org/search.html> for the error or if that fails, try searching Google. Often, a distribution will have already solved the problem (many of them use development versions of packages, so they see the changes sooner than those of us who normally use stable released versions). But be cautious - all builders tend to carry patches which are no longer necessary, and to have fixes which are only required because of their particular choices in how they build a package. You may have to search deeply to find a fix for the package version you are trying to use, or even to find the package (names are sometimes not what you might expect, e.g. ghostscript often has a prefix or a suffix in its name), but the following notes might help, particularly for those who, like the editors, are trying to build the latest versions and encountering problems:
 - Arch <https://www.archlinux.org/packages/> - enter the package name in the 'Keywords' box, select the package name, select the 'Source Files' field, and then select the `PKGBUILD` entry to see how they build this package.
 - Debian <http://ftp.debian.org/debian/pool> (use your country's version if there is one) - the source will be in `.tar.gz` tarballs (either the original upstream `.orig` source, or else a `dfsg` containing those parts which comply with Debian's free software guidelines) accompanied by versioned `.diff.gz` or `.tar.gz` additions. These additions often show how the package is built, and may contain patches. In the `.diff.gz` versions, any patches create files in `debian/patches`.
 - Fedora package source gets reorganized from time to time. At the moment the package source for rpms is at <https://src.fedoraproject.org/projects/rpms/%2A> and from there you can try putting a package name in the search box. If the package is found you can look at the files (specfile to control the build, various patches) or the commits. If that fails, you can download an srpm (source rpm) and using `rpm2cpio` (see the Tip at the bottom of the page). For rpms go to <https://dl.fedoraproject.org/pub/fedora/linux/> and then choose which repo you wish to look at - development/rawhide is the latest development, or choose releases for what was shipped in a release, updates for updates to a release, or updates/testing for the latest updates which might work or might have problems.
 - Gentoo - First use a search engine to find an ebuild which looks as if it will fix the problem, or search at <https://packages.gentoo.org/> - use the search field. Note where the package lives in the portage hierarchy, e.g. `app-something/`. In general you can treat the ebuild as a sort of pseudo-code / shell combination with some functions you can hazard a guess at, such as `dodoc`. If the fix is just a `sed`, try it. However, in most cases the fix will use a patch. To find the patch, use a gentoo-portage mirror: Two links to mirrors in the U.S.A. which seem to usually be up to date are <https://mirror.rackspace.com/gentoo-portage/> and <https://mirror.steadfast.net/gentoo-portage/>. Navigate down the tree to the package, then to the `files/` directory to look for the patch. Sometimes a portage mirror has not yet been updated, particularly for a recent new patch. In a few cases, gentoo batch the patches into a tarball and the ebuild will have a link in the form `https://dev.gentoo.org/~${PATCH_DEV}/distfiles/${P}-patches-${PATCH_VER}.tar.xz` here, look for `PATCH_DEV` and `PATCH_VER` in the build and format the full URL in your browser or for wget. Remember the '`~`' before the developer's ID and note that trying to search the earlier levels of the URL in a browser may drop you at www.gentoo.org or return 403 (forbidden).
 - openSUSE provide a rolling release, some package versions are in <https://download.opensuse.org/source/tumbleweed/repo/oss/src/> but others are in `..//update/openSUSE-current/src` - the source only seems to be available in source rpms.
 - Slackware - the official package browser is currently broken. The site at <https://slackbuilds.org/> has current and previous versions in their unofficial repository with links to homepages, downloads, and some individual files, particularly the `.slackBuild` files.
 - Ubuntu <http://ftp.ubuntu.com/ubuntu/pool/> - see the Debian notes above.

If everything else fails, try the blfs-support mailing-list.

Tip

If you have found a package that is only available in `.deb` or `.rpm` format, there are two small scripts, `rpm2targz` and `deb2targz` that are available at <https://anduin.linuxfromscratch.org/BLFS/extras/deb2targz.tar.bz2> and <https://anduin.linuxfromscratch.org/BLFS/extras/rpm2targz.tar.bz2> to convert the archives into a simple `.tar.gz` format.

You may also find an `rpm2cpio` script useful. The Perl version in the linux kernel archives at <https://lore.kernel.org/all/20021016121842.GA2292@ncsu.edu/2-rpm2cpio> works for most source rpms. The `rpm2targz` script will use an `rpm2cpio` script or binary if one is on your path. Note that `rpm2cpio` will unpack a source rpm in the current directory, giving a tarball, a spec file, and perhaps patches or other files.

Part II. Post LFS Configuration and Extra Software

Chapter 3. After LFS Configuration Issues

The intention of LFS is to provide a basic system which you can build upon. There are several things about tidying up the system which many people wonder about once they have done the base install. We hope to cover these issues in this chapter.

Most people coming from non-Unix like backgrounds to Linux find the concept of text-only configuration files slightly strange. In Linux, just about all configuration is done via the manipulation of text files. The majority of these files can be found in the `/etc` hierarchy. There are often graphical configuration programs available for different subsystems but most are simply pretty front ends to the process of editing a text file. The advantage of text-only configuration is that you can edit parameters using your favorite text editor, whether that be `vim`, `emacs`, or any other editor.

The first task is making a recovery boot device in [Creating a Custom Boot Device](#) because it's the most critical need. Hardware issues relevant to firmware and other devices is addressed next. The system is then configured to ease addition of new users, because this can affect the choices you make in the two subsequent topics—[The Bash Shell Startup Files](#) and [The vimrc Files](#).

There is one remaining topic: [Customizing your Logon with /etc/issue](#). It doesn't have much interaction with the other topics in this chapter.

Creating a Custom Boot Device

Decent Rescue Boot Device Needs

This section is really about creating a *rescue* device. As the name *rescue* implies, the host system has a problem, often lost partition information or corrupted file systems, that prevents it from booting and/or operating normally. For this reason, you *must not* depend on resources from the host being "rescued". To presume that any given partition or hard drive *will* be available is a risky presumption.

In a modern system, there are many devices that can be used as a rescue device: floppy, cdrom, usb drive, or even a network card. Which one you use depends on your hardware and your BIOS. In the past, a rescue device was thought to be a floppy disk. Today, many systems do not even have a floppy drive.

Building a complete rescue device is a challenging task. In many ways, it is equivalent to building an entire LFS system. In addition, it would be a repetition of information already available. For these reasons, the procedures for a rescue device image are not presented here.

Creating a Rescue Floppy

The software of today's systems has grown large. Linux 2.6 no longer supports booting directly from a floppy. In spite of this, there are solutions available using older versions of Linux. One of the best is Tom's Root/Boot Disk available at <http://www.toms.net/rb/>. This will provide a minimal Linux system on a single floppy disk and provides the ability to customize the contents of your disk if necessary.

Creating a Bootable CD-ROM

There are several sources that can be used for a rescue CD-ROM. Just about any commercial distribution's installation CD-ROMs or DVDs will work. These include RedHat, Ubuntu, and SuSE. One very popular option is Knoppix.

Also, the LFS Community has developed its own LiveCD available at <https://www.linuxfromscratch.org/livecd/>. This LiveCD, is no longer capable of building an entire LFS/BLFS system, but is still a good rescue CD-ROM. If you download the ISO image, use `xorriso` to copy the image to a CD-ROM.

The instructions for using GRUB2 to make a custom rescue CD-ROM are also available in [LFS Chapter 10](#).

Creating a Bootable USB Drive

A USB Pen drive, sometimes called a Thumb drive, is recognized by Linux as a SCSI device. Using one of these devices as a rescue device has the advantage that it is usually large enough to hold more than a minimal boot image. You can save critical data to the drive as well as use it to diagnose and recover a damaged system. Booting such a drive requires BIOS support, but building the system consists of formatting the drive, adding GRUB as well as the Linux kernel and supporting files.

About Console Fonts

An LFS system can be used without a graphical desktop, and unless or until you install [a graphical environment](#) you will have to work in the console. Most, if not all, PCs boot with an 8x16 font - whatever the actual screen size. There are a few things you can do to alter the display on the console. Most of them involve changing the font, but the first alters the commandline used by grub.

Setting a smaller screen size in grub

Modern screens often have a lot more pixels than the screens used in the past. If your screen is 1600 pixels wide, an 8x16 font will give you 200 columns of text - unless your monitor is enormous, the text will be tiny. One of the ways to work around this is to tell grub to use a smaller size, such as 1024x768 or 800x600 or even 640x480. Even if your screen does not have a 4:3 aspect ratio, this should work.

To try this, you can reboot and edit grub's command-line to insert a 'video=' parameter between the 'root=/dev/sdXn' and 'ro', for example `root=/dev/sda2 video=1024x768 ro` based on the example in LFS section 10.4.4 : [..../..../lfs/view/12.2-systemd/chapter10/grub.html](#).

If you decide that you wish to do this, you can then (as the `root` user) edit `/boot/grub/grub.cfg`.

Using the standard psf fonts

In LFS the kbd package is used. The fonts it provides are PC Screen Fonts, usually called PSF, and they were installed into `/usr/share/consolefonts`. Where these include a unicode mapping table, the file suffix is often changed to `.psfu` although packages such as terminus-font (see below) do not add the 'u'. These fonts are usually compressed with gzip to save space, but that is not essential.

The initial PC text screens had 8 colours, or 16 colours if the bright versions of the original 8 colours were used. A PSF font can include up to 256 characters (technically, glyphs) while allowing 16 colours, or up to 512 characters (in which case, the bright colours will not be available). Clearly, these console fonts cannot be used to display CJK text - that would need thousands of available glyphs.

Some fonts in kbd can cover more than 512 codepoints ('characters'), with varying degrees of fidelity: unicode contains several whitespace codepoints which can all be mapped to a space, varieties of dashes can be mapped to a minus sign, smart quotes can map to the regular ASCII quotes rather than to whatever is used for "codepoint not present or invalid", and those cyrillic or greek letters which look like latin letters can be mapped onto them, so 'A' can also do duty for cyrillic A and greek Alpha, and 'P' can also do duty for cyrillic ER and greek RHO. Unfortunately, where a font has been created from a BDF file (the method in terminus and Debian's [console-setup](#)) such mapping of additional codepoints onto an existing glyph is not always done, although the terminus ter-vXXn fonts do this well.

There are over 120 combinations of font and size in kbd: often a font is provided at several character sizes, and sometimes varieties cover different subsets of unicode. Most are 8 pixels wide, in heights from 8 to 16 pixels, but there are a few which are 9 pixels wide, some others which are 12x22, and even one (`latarcyrheb-sun32.psfu`) which has been scaled up to 16x32. Using a bigger font is another way of making text on a large screen easier to read.

Testing different fonts

You can test fonts as a normal user. If you have a font which has not been installed, you can load it with :

```
setfont /path/to/yourfont.ext
```

For the fonts already installed you only need the name, so using `gr737a-9x16.psfu.gz` as an example:

```
setfont gr737a-9x16
```

To see the glyphs in the font, use:

```
showconsolefont
```

If the font looks as if it might be useful, you can then go on to test it more thoroughly.

When you find a font which you wish to use, as the `root` user) edit `/etc/vconsole.conf` as described in LFS section 9.6 [..../..../lfs/view/12.2-systemd/chapter09/console.html](#).

For fonts not supplied with the kbd package you will need to optionally compress it / them with `gzip` and then install it / them as the `root` user.

Editing fonts using psf-tools

Although some console fonts are created from BDF files, which is a text format with hex values for the pixels in each row of the character, there are more-modern tools available for editing psf fonts. The `psf-tools` package allows you to dump a font to a text representation with a dash for a pixel which is off (black) and a hash for a pixel which is on (white). You can then edit the text file to add more characters, or reshape them, or map extra codepoints onto them, and then create a new psf font with your changes.

Using fonts from Terminus-font

The [Terminus Font](#) package provides fixed-width bitmap fonts designed for long (8 hours and more per day) work with computers. Under 'Character variants' on that page is a list of patches (in the `alt/` directory). If you are using a graphical browser to look at that page, you can see what the patches do, e.g. 'I12' makes 'I' more visibly different from 'i' and '1'.

By default terminus-fonts will try to create several types of font, and it will fail if `bdf2opcf` from [Xorg Applications](#) has not been installed. The configure script is only really useful if you go on to install *all* the fonts (console and X11 bitmap) to the correct directories, as in a distro. To build only the PSF fonts and their dependencies, run:

```
make psf
```

This will create more than 240 ter-*.psf fonts. The 'b' suffix indicates bright, 'n' indicates normal. You can then test them to see if any fit your requirements. Unless you are creating a distro, there seems little point in installing them all.

As an example, to install the last of these fonts, you can gzip it and then as the `root` user:

```
install -v -m644 ter-v32n.psf.gz /usr/share/consolefonts
```

About Firmware

On some recent PCs it can be necessary, or desirable, to load firmware to make them work at their best. There is a directory, `/lib/firmware`, where the kernel or kernel drivers look for firmware images.

Currently, most firmware can be found at a `git` repository which can be viewed in the browser with the URL <https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git/plain>. For convenience, the LFS Project has created a mirror, updated daily, where these firmware files can be accessed via `wget` or a web browser at <https://anduin.linuxfromscratch.org/BLFS/linux-firmware/>.

To get the firmware, point a browser to one of the above repositories and then download the item(s) which you need. If you want all these firmware files (for example you are distributing the system onto multiple hardware systems), either install [git-2.46.0](#) and clone <https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git>, or open this URL in a browser and download the latest snapshot listed in the `Tag` table.

For some other firmware, particularly for Intel microcode and certain wifi devices, the needed firmware is not available in the above repository. Some of this will be addressed below, but a search of the Internet for needed firmware is sometimes necessary.

Firmware files are conventionally referred to as blobs because you cannot determine what they will do. Note that firmware is distributed under various different licenses which do not permit disassembly or reverse-engineering.

Firmware for PCs falls into four categories:

- Updates to the CPU to work around errata, usually referred to as microcode.
- Firmware for video controllers. On x86 machines this is required for ATI devices (Radeon and AMDGPU chips) and may be useful for Intel (Skylake and later) and Nvidia (Kepler and later) GPUs.

ATI Radeon and AMDGPU devices all require firmware to be able to use KMS (kernel modesetting - the preferred option) as well as for Xorg. For old radeon chips (before the R600), the firmware is still in the kernel source.

Intel integrated GPUs from Skylake onwards can use firmware for GuC (the Graphics microcontroller), and also for the HuC (HEVC/H265 microcontroller which offloads to the GPU) and the DMC (Display Microcontroller) to provide additional low-power states. The GuC and HuC have had a chequered history in the kernel and updated firmware may be disabled by default, depending on your kernel version. Further details may be found at [01.org](#) and [Arch linux](#).

Nvidia GPUs from Kepler onwards require signed firmware, otherwise the nouveau driver is unable to provide hardware acceleration. Nvidia has now released firmware up to Ampere (GeForce30 series) to linux-firmware. Note that faster clocks than the default are not enabled by the released firmware.

- Firmware updates for wired network ports. Most of them work even without the updates, but they will probably work better with the updated firmware. For some modern laptops, firmware for both wired ethernet (e.g. `rtl_nic`) and also for bluetooth devices (e.g. `qca`) is *required* before the wired network can be used.
- Firmware for other devices, such as wireless NICs. These devices are not required for the PC to boot, but need the firmware before these devices can be used.

Note

Although not needed to load a firmware blob, the following tools may be useful for determining, obtaining, or preparing the needed firmware in order to load it into the system: [cpio-2.15](#), [git-2.46.0](#), [pciutils-3.13.0](#), and [Wget-1.24.5](#)

Microcode updates for CPUs

In general, microcode can be loaded by the BIOS or UEFI, and it might be updated by upgrading to a newer version of those. On linux, you can also load the microcode from the kernel if you are using an AMD family 10h or later processor (first introduced late 2007), or an Intel processor from 1998 and later (Pentium4, Core, etc), if updated microcode has been released. These updates only last until the machine is powered off, so they need to be applied on every boot.

Intel provide updates of their microcode for Skylake and later processors as new vulnerabilities come to light, and have in the past provided updates for processors from SandyBridge onwards, although those are no-longer supported for new fixes. New versions of AMD firmware are rare and usually only apply to a few models, although motherboard manufacturers get

AGESA (AMD Generic Encapsulated Software Architecture) updates to change BIOS values, e.g. to support more memory variants, new vulnerability fixes or newer CPUs.

There were two ways of loading the microcode, described as 'early' and 'late'. Early loading happens before userspace has been started, late loading happens after userspace has started. However, late loading is known to be problematic and not supported anymore (see the kernel commit [x86/microcode: Taint and warn on late loading](#)). Indeed, early loading is needed to work around one particular erratum in early Intel Haswell processors which had TSX enabled. (See [Intel Disables TSX Instructions: Erratum Found in Haswell, Haswell-E/EP, Broadwell-Y](#).) Without this update glibc can do the wrong thing in uncommon situations.

In previous versions of this book, late loading of microcode to see if it gets applied was recommended, followed by using an initrd to force early loading. But now that the contents of the Intel microcode tarball is documented, and AMD microcode can be read by a Python script to determine which machines it covers, there is no real reason to use late loading.

It might be still possible to manually force late loading of microcode. But it may cause kernel malfunction and you should take the risk yourself. You will need to reconfigure your kernel for late loading, but early loading is always supported by Linux kernel version 6.6 or later on a x86 (no matter 32-bit or 64-bit) system. The instructions here will show you how to create an initrd for early loading. It is also possible to build the same microcode bin file into the kernel, which allows early loading but requires the kernel to be recompiled to update the microcode.

To confirm what processor(s) you have (if more than one, they will be identical) look in /proc/cpuinfo. Determine the decimal values of the cpu family, model and stepping by running the following command (it will also report the current microcode version):

```
head -n7 /proc/cpuinfo
```

Convert the cpu family, model and stepping to pairs of hexadecimal digits, and remember the value of the "microcode" field. You can now check if there is any microcode available.

If you are creating an initrd to update firmware for different machines, as a distro would do, go down to 'Early loading of microcode' and cat all the Intel blobs to GenuineIntel.bin or cat all the AMD blobs to AuthenticAMD.bin. This creates a larger initrd - for all Intel machines in the 20200609 update the size was 3.0 MB compared to typically 24 KB for one machine.

Intel Microcode for the CPU

The first step is to get the most recent version of the Intel microcode. This must be done by navigating to <https://github.com/intel/Intel-Linux-Processor-Microcode-Data-Files/releases/> and downloading the latest file there. As of this writing the most secure version of the microcode is microcode-20240813. Extract this file in the normal way, the microcode is in the intel-ucode directory, containing various blobs with names in the form XX-YY-ZZ. There are also various other files, and a release note.

In the past, intel did not provide any details of which blobs had changed versions, but now the release note details this. You can compare the microcode version in /proc/cpuinfo with the version for your CPU model in the releases note to know if there is an update.

The recent firmware for older processors is provided to deal with vulnerabilities which have now been made public, and for some of these such as Microarchitectural Data Sampling (MDS) you might wish to increase the protection by disabling hyperthreading, or alternatively to disable the kernel's default mitigation because of its impact on compile times. Please read the online documentation at <https://www.kernel.org/doc/html/latest/admin-guide/hw-vuln/index.html>.

For an Tigerlake mobile (described as Intel(R) Core(TM) i5-11300H CPU) the relevant values are cpu family 6, model 140, stepping 1 so in this case the required identification is 06-8c-01. The releases note says the latest microcode for it is versioned 0xb8. If the value of the "microcode" field in /proc/cpuinfo is 0xb8 or greater, it indicates the microcode update is already applied by the BIOS. Otherwise, proceed to [the section called "Early loading of microcode"](#).

AMD Microcode for the CPU

Begin by downloading a container of firmware for your CPU family from <https://anduin.linuxfromscratch.org/BLFS/linux-firmware/amd-ucode/>. The family is always specified in hex. Families 10h to 14h (16 to 20) are in microcode_amd.bin. Families 15h, 16h, 17h (Zen, Zen+, Zen2) and 19h (Zen3) have their own containers, but very few machines are likely to get updated microcode. Instead, AMD provide an updated AGESA to the motherboard makers, who may provide an updated BIOS using this. There is a Python3 script at https://github.com/AMDESE/amd_ucode_info/blob/master/amd_ucode_info.py. Download that script and run it against the bin file to check which processors have updates.

For the very old Athlon(tm) II X2 in these examples the values were cpu family 16, model 5, stepping 3 giving an identification of Family=0x10 Model=0x05 Stepping=0x03. One line of the amd_ucode_info.py script output describes the microcode version for it:

```
Family=0x10 Model=0x05 Stepping=0x03: Patch=0x010000c8 Length=960 bytes
```

If the value of the "microcode" field in /proc/cpuinfo is 0x10000c8 or greater, it indicates the BIOS has already applied the microcode update. Otherwise, proceed to [the section called "Early loading of microcode"](#).

Early loading of microcode

If you have established that updated microcode is available for your system, it is time to prepare it for early loading. This requires an additional package, [cpio-2.15](#) and the creation of an initrd which will need to be added to grub.cfg.

It does not matter where you prepare the initrd, and once it is working you can apply the same initrd to later LFS systems or newer kernels on this same machine, at least until any newer microcode is released. Use the following commands:

```
mkdir -p initrd/kernel/x86/microcode  
cd initrd
```

For an AMD machine, use the following command (replace <MYCONTAINER> with the name of the container for your CPU's family):

```
cp -v ../<MYCONTAINER> kernel/x86/microcode/AuthenticAMD.bin
```

Or for an Intel machine copy the appropriate blob using this command:

```
cp -v ../../intel-ucode/<XX-YY-ZZ> kernel/x86/microcode/GenuineIntel.bin
```

Now prepare the initrd:

```
find . | cpio -o -H newc > /boot/microcode.img
```

You now need to add a new entry to /boot/grub/grub.cfg and here you should add a new line after the linux line within the stanza. If /boot is a separate mountpoint:

```
initrd /microcode.img
```

or this if it is not:

```
initrd /boot/microcode.img
```

If you are already booting with an initrd (see [the section called "About initramfs"](#)), you should run `mkinitramfs` again after putting the appropriate blob or container into `/lib/firmware`. More precisely, put an intel blob in a `/lib/firmware/intel-ucode` directory or an AMD container in a `/lib/firmware/amd-ucode` directory before running `mkinitramfs`. Alternatively, you can have both initrd on the same line, such as `initrd /microcode.img /other-initrd.img` (adapt that as above if /boot is not a separate mountpoint).

You can now reboot with the added initrd, and then use the following command to check that the early load worked:

```
dmesg | grep -e 'microcode' -e 'Linux version' -e 'Command line'
```

If you updated to address vulnerabilities, you can look at the output of the `lscpu` command to see what is now reported.

The places and times where early loading happens are very different in AMD and Intel machines. First, an example of an Intel (Tigerlake mobile) with early loading:

```
[ 0.000000] Linux version 6.10.4 (xry111@stargazer) (gcc (GCC) 14.2.0, GNU ld (GNU Binutils) 2.43) #4 SMP PREEMPT_DYNAMIC Tue Aug  
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-6.10.0 root=PARTUUID=<CLASSIFIED> ro  
[ 0.585605] microcode: Current revision: 0x000000b8  
[ 0.585611] microcode: Updated early from: 0x00000086
```

A historic AMD example:

```
[ 0.000000] Linux version 4.15.3 (ken@testserver) (gcc version 7.3.0 (GCC))  
#2 SMP Sun Feb 18 02:32:03 GMT 2018  
[ 0.000000] Command line: BOOT_IMAGE=/vmlinuz-4.15.3-sda5 root=/dev/sda5 ro  
[ 0.307619] microcode: microcode updated early to new patch_level=0x010000c8  
[ 0.307678] microcode: CPU0: patch_level=0x010000c8  
[ 0.307723] microcode: CPU1: patch_level=0x010000c8  
[ 0.307795] microcode: Microcode Update Driver: v2.2.
```

Firmware for Video Cards

Firmware for ATI video chips (R600 and later)

These instructions do NOT apply to old radeons before the R600 family. For those, the firmware is in the kernel's `/lib/firmware/` directory. Nor do they apply if you intend to avoid a graphical setup such as Xorg and are content to use the default 80x25 display rather than a framebuffer.

Early radeon devices only needed a single 2K blob of firmware. Recent devices need several different blobs, and some of them are much bigger. The total size of the radeon firmware directory is over 500K — on a large modern system you can

probably spare the space, but it is still redundant to install all the unused files each time you build a system.

A better approach is to install [pciutils-3.13.0](#) and then use `lspci` to identify which VGA controller is installed.

With that information, check the RadeonFeature page of the Xorg wiki for [Decoder ring for engineering vs marketing names](#) to identify the family (you may need to know this for the Xorg driver in BLFS — Southern Islands and Sea Islands use the radeonsi driver) and the specific model.

Now that you know which controller you are using, consult the [Radeon](#) page of the Gentoo wiki which has a table listing the required firmware blobs for the various chipsets. Note that Southern Islands and Sea Islands chips use different firmware for kernel 3.17 and later compared to earlier kernels. Identify and download the required blobs then install them:

```
mkdir -pv /lib/firmware/radeon
cp -v <YOUR_BLOBS> /lib/firmware/radeon
```

Building the kernel amdgpu driver as a module is recommended because the firmware files need to be accessible at the time it is loaded. If you are building it as a part of the kernel image for any reason, you need to either include the firmware files in the initramfs (read [the section called "About initramfs"](#) for details), or include them in the kernel image itself (read [the section called "Include Firmware Blobs in the Kernel Image"](#) for details).

Firmware for AMD/ATI amdgpu video chips

All video controllers using the amdgpu kernel driver require firmware, whether you will be using the xorg amdgpu driver, the xserver's modesetting driver, or just kernel modesetting to get a console framebuffer larger than 80x25.

Install [pciutils-3.13.0](#) and use that to check the model name (look for 'VGA compatible controller:'). If you have an APU (Accelerated Processing Unit, i.e. CPU and video on the same chip) that will probably tell you the name. If you have a separate amdgpu video card you will need to search to determine which name it uses (e.g. a card described as Advanced Micro Devices, Inc. [AMD/ATI] Baffin [Radeon RX 550 640SP / RX 560/560X] needs Polaris11 firmware. There is a table of "Family, Chipset name, Product name and Firmware" at the end of the Kernel sections in [AMDGPU](#) page of the Gentoo wiki.

Once you have identified the firmware name, install all the relevant files for it. For example, the Baffin card mentioned above has 21 different polaris11* files, APUs such as renoir and picasso have at least 12 files and might gain more in future updates (e.g. the raven APU now has a 13th file, raven_ta.bin).

```
mkdir -pv /lib/firmware/amdgpu
cp -v <YOUR_BLOBS> /lib/firmware/amdgpu
```

If disk space is not a problem, you could install all the current amdgpu firmware files and not worry about exactly which chipset is installed.

Building the kernel amdgpu driver as a module is recommended because the firmware files need to be accessible at the time it is loaded. If you are building it as a part of the kernel image for any reason, you need to either include the firmware files in the initramfs (read [the section called "About initramfs"](#) for details), or include them in the kernel image itself (read [the section called "Include Firmware Blobs in the Kernel Image"](#) for details).

Firmware for Nvidia video chips

Nvidia has released basic signed firmware for recent graphics chips, but significantly after the chips and its own binary drivers were first available. For other chips it has been necessary to extract the firmware from the binary driver.

For more exact information about which chips need extracted firmware, see <https://nouveau.freedesktop.org/VideoAcceleration.html>.

If the necessary firmware is available in the `nvidia/` directory of `linux-firmware`, copy it to `/lib/firmware/nouveau`.

If the firmware has not been made available in `linux-firmware`, for the old chips mentioned in the nouveau wiki link above run the following commands:

```
wget https://anduin.linuxfromscratch.org/BLFS/nvidia-firmware/extract_firmware.py
wget https://us.download.nvidia.com/XFree86/Linux-x86/340.32/NVIDIA-Linux-x86-340.32.run
sh NVIDIA-Linux-x86-340.32.run --extract-only
python3 extract_firmware.py
mkdir -p /lib/firmware/nouveau
cp -d nv* vuc-* /lib/firmware/nouveau/
```

Firmware for Network Interfaces

The kernel likes to load firmware for some network drivers, particularly those from Realtek (the `/lib/linux-firmware/rtl_nic/`) directory, but they generally appear to work without it. Therefore, you can boot the kernel, check `dmesg` for messages about this missing firmware, and if necessary download the firmware and put it in the specified directory in `/lib/firmware` so that it will be found on subsequent boots. Note that with current kernels this works whether or not the driver is compiled in or built as a module, there is no need to build this firmware into the kernel. Here is an example where the R8169 driver has been compiled in but the firmware was not made available. Once the firmware had been provided, there was no mention of it on later boots.

```
dmesg | grep firmware | grep r8169
[    7.018028] r8169 0000:01:00.0: Direct firmware load for rtl_nic/rtl18168g-2.fw failed with error -2
[    7.018036] r8169 0000:01:00.0 eth0: unable to load firmware patch rtl_nic/rtl18168g-2.fw (-2)
```

Firmware for Regulatory Database of Wireless Devices

Different countries have different regulations on the radio spectrum usage of wireless devices. You can install a firmware to make the wireless devices obey local spectrum regulations, so you won't be inquired by local authority or find your wireless NIC jamming the frequencies of other devices (for example, remote controllers). The regulatory database firmware can be downloaded from <https://kernel.org/pub/software/network/wireless-regdb/>. To install it, simply extract `regulatory.db` and `regulatory.db.p7s` from the tarball into `/lib/firmware`. Note that either the `cfg80211` driver needs to be selected as a module for the `regulatory.*` files to be loaded, or those files need to be included as firmware into the kernel, as explained above in [the section called "Firmware for Video Cards"](#).

The access point (AP) would send a country code to your wireless NIC, and [wpa_supplicant-2.11](#) would tell the kernel to load the regulation of this country from `regulatory.db`, and enforce it. Note that several AP don't send this country code, so you may be locked to a rather restricted usage (specially if you want to use your interface as an AP).

Sound Open Firmware

Some systems (especially budget laptops) utilize a DSP shipped with the CPU for connection with the audio codec. The Sound Open Firmware must be loaded onto the DSP to make it functional. These firmware files can be downloaded from <https://github.com/thesofproject/sof-bin/releases>. Extract the tarball and changing into the extracted directory, then as the `root` user install the firmware:

```
install -vdm755 /usr/lib/firmware/intel    &&
cp -av -T --no-preserve=ownership sof      \
  /usr/lib/firmware/intel/sof               &&
cp -av -T --no-preserve=ownership sof-tplg \
  /usr/lib/firmware/intel/sof-tplg
```

[alsa-lib-1.2.12](#) needs Use Case Manager configuration files for the systems using Sound Open Firmware as well. Read the [alsa-lib-1.2.12](#) page for the instructions to install them. Once the firmware is loaded (you may need a reboot so the kernel will load them) and the UCM configuration files are installed, following [the section called "Configuring ALSA Utilities"](#) to set up your sound card for ALSA properly.

Firmware for Other Devices

Identifying the correct firmware will typically require you to install [pciutils-3.13.0](#), and then use `lspci` to identify the device. You should then search online to check which module it uses, which firmware, and where to obtain the firmware — not all of it is in `linux-firmware`.

If possible, you should begin by using a wired connection when you first boot your LFS system. To use a wireless connection you will need to use a network tools such as [iw-6.9](#), [Wireless Tools-29](#), or [wpa_supplicant-2.11](#).

Firmware may also be needed for other devices such as some SCSI controllers, bluetooth adaptors, or TV recorders. The same principles apply.

Include Firmware Blobs in the Kernel Image

Some drivers, notably the drivers for ATI or AMD GPU, requires the firmware files accessible at the time it is loaded. The easiest method to handle these drivers is building them as a kernel module. An alternative method is creating an initramfs (read [the section called "About initramfs"](#) for details) including the firmware files. If you don't want to use either methods, you may include the firmware files in the kernel image itself. Install the needed firmware files into `/lib/firmware` first, then set the following kernel configuration and rebuild the kernel:

```
Device Drivers --->
  Generic Driver Options --->
    Firmware loader --->
      <*>           Firmware loading facility          [FW_LOADER]
      (xx/aa.bin xx/bb.bin) Build named firmware blobs into the kernel binary
                           ... [EXTRA_FIRMWARE]
      (/lib/firmware)   Firmware blobs root directory
                           ... [EXTRA_FIRMWARE_DIR]
```

Replace `xx/aa.bin` `xx/bb.bin` with a whitespace-separated list of paths to the needed firmware files, relative to `/lib/firmware`. A method easier than manually typing the list (it may be long) is running the following command:

```
echo CONFIG_EXTRA_FIRMWARE='$(cd /lib/firmware; echo amdgpu/*)' >> .config
make oldconfig
```

Replace `amdgpu/*` with a shell pattern matching the needed firmware files.

Warning

Do not distribute a kernel image containing the firmware to others or you may violate the GPL.

About Devices

Although most devices needed by packages in BLFS and beyond are set up properly by udev using the default rules installed by LFS in `/etc/udev/rules.d`, there are cases where the rules must be modified or augmented.

Multiple Sound Cards

If there are multiple sound cards in a system, the "default" sound card becomes random. The method to establish sound card order depends on whether the drivers are modules or not. If the sound card drivers are compiled into the kernel, control is via kernel command line parameters in `/boot/grub/grub.cfg`. For example, if a system has both an FM801 card and a SoundBlaster PCI card, the following can be appended to the command line:

```
snd-fm801.index=0 snd-ens1371.index=1
```

If the sound card drivers are built as modules, the order can be established in the `/etc/modprobe.conf` file with:

```
options snd-fm801 index=0
options snd-ens1371 index=1
```

USB Device Issues

USB devices usually have two kinds of device nodes associated with them.

The first kind is created by device-specific drivers (e.g., `usb_storage/sd_mod` or `usbip`) in the kernel. For example, a USB mass storage device would be `/dev/sdb`, and a USB printer would be `/dev/usb/lp0`. These device nodes exist only when the device-specific driver is loaded.

The second kind of device nodes (`/dev/bus/usb/BBB/DDD`, where BBB is the bus number and DDD is the device number) are created even if the device doesn't have a kernel driver. By using these "raw" USB device nodes, an application can exchange arbitrary USB packets with the device, i.e., bypass the possibly-existing kernel driver.

Access to raw USB device nodes is needed when a userspace program is acting as a device driver. However, for the program to open the device successfully, the permissions have to be set correctly. By default, due to security concerns, all raw USB devices are owned by user root and group root, and have 0664 permissions (the read access is needed, e.g., for `lsusb` to work and for programs to access USB hubs). Packages (such as `SANE` and `libgphoto2`) containing userspace USB device drivers also ship udev rules that change the permissions of the controlled raw USB devices. That is, rules installed by `SANE` change permissions for known scanners, but not printers. If a package maintainer forgot to write a rule for your device, report a bug to both BLFS (if the package is there) and upstream, and you will need to write your own rule.

Before Linux-2.6.15, raw USB device access was performed not with `/dev/bus/usb/BBB/DDD` device nodes, but with `/proc/bus/usb/BBB/DDD` pseudofiles. Some applications still use only this deprecated technique and can't use the new device nodes. They cannot work with Linux kernel version 3.5 or newer. If you need to run such an application, contact the developer of it for a fix.

Udev Device Attributes

Fine-tuning of device attributes such as group name and permissions is possible by creating extra udev rules, matching on something like this. The vendor and product can be found by searching the `/sys/devices` directory entries or using `udevadm info` after the device has been attached. See the documentation in the current udev directory of `/usr/share/doc` for details.

```
SUBSYSTEM=="usb_device", SYSFS{idVendor}=="05d8", SYSFS{idProduct}=="4002", \
GROUP=="scanner", MODE=="0660"
```

Note

The above line is used for descriptive purposes only. The scanner udev rules are put into place when installing [SANE-1.2.1](#).

Devices for DVD Drives

If the initial boot process does not set up the `/dev/dvd` device properly, it can be installed using the following modification to the default udev rules. As the `root` user, run:

```
sed '1d;/SYMLINK.*cdrom/ a\  
KERNEL=="sr0", ENV{ID_CDROM_DVD}=="1", SYMLINK+="dvd", OPTIONS+="link_priority=-100" \  
/lib/udev/rules.d/60-cdrom_id.rules > /etc/udev/rules.d/60-cdrom_id.rules
```

Configuring for Adding Users

Together, the `/usr/sbin/useradd` command and `/etc/skel` directory (both are easy to set up and use) provide a way to assure new users are added to your LFS system with the same beginning settings for things such as the `PATH`, keyboard processing and other environmental variables. Using these two facilities makes it easier to assure this initial state for each new user added to the system.

The `/etc/skel` directory holds copies of various initialization and other files that may be copied to the new user's home directory when the `/usr/sbin/useradd` program adds the new user.

Useradd

The `useradd` program uses a collection of default values kept in `/etc/default/useradd`. This file is created in a base LFS installation by the Shadow package. If it has been removed or renamed, the `useradd` program uses some internal defaults. You can see the default values by running `/usr/sbin/useradd -D`.

To change these values, simply modify the `/etc/default/useradd` file as the `root` user. An alternative to directly modifying the file is to run `useradd` as the `root` user while supplying the desired modifications on the command line. Information on how to do this can be found in the `useradd` man page.

/etc/skel

To get started, create an `/etc/skel` directory and make sure it is writable only by the system administrator, usually `root`. Creating the directory as `root` is the best way to go.

The mode of any files from this part of the book that you put in `/etc/skel` should be writable only by the owner. Also, since there is no telling what kind of sensitive information a user may eventually place in their copy of these files, you should make them unreadable by "group" and "other".

You can also put other files in `/etc/skel` and different permissions may be needed for them.

Decide which initialization files should be provided in every (or most) new user's home directory. The decisions you make will affect what you do in the next two sections, [The Bash Shell Startup Files](#) and [The vimrc Files](#). Some or all of those files will be useful for `root`, any already-existing users, and new users.

The files from those sections that you might want to place in `/etc/skel` include `.inputrc`, `.bash_profile`, `.bashrc`, `.bash_logout`, `.dircolors`, and `.vimrc`. If you are unsure which of these should be placed there, just continue to the following sections, read each section and any references provided, and then make your decision.

You will run a slightly modified set of commands for files which are placed in `/etc/skel`. Each section will remind you of this. In brief, the book's commands have been written for files *not* added to `/etc/skel` and instead just sends the results to the user's home directory. If the file is going to be in `/etc/skel`, change the book's command(s) to send output there instead and then just copy the file from `/etc/skel` to the appropriate directories, like `/etc`, `~` or the home directory of any other user already in the system.

When Adding a User

When adding a new user with `useradd`, use the `-m` parameter, which tells `useradd` to create the user's home directory and copy files from `/etc/skel` (can be overridden) to the new user's home directory. For example (perform as the `root` user):

```
useradd -m <newuser>
```

If you are sharing a `/home` or `/usr/src` with another Linux distro (for example, the host distro used for building LFS), you can create a user with the same UID (and, same primary group GID) to keep the file ownership consistent across the systems. First, on *the other distro*, get the UID of the user and the GID of the user's primary group:

```
getent passwd <username> | cut -d ':' -f 3,4
```

The command should output the UID and GID, separated by a colon. Now on the BLFS system, create the primary group and the user:

```
groupadd -g <GID> <username> &&
useradd -u <UID> -g <username> <username>
```

About System Users and Groups

Throughout BLFS, many packages install programs that run as daemons or in some way should have a user or group name assigned. Generally these names are used to map a user ID (uid) or group ID (gid) for system use. Generally the specific uid or gid numbers used by these applications are not significant. The exception of course, is that `root` has a uid and gid of 0 (zero) that is indeed special. The uid values are stored in `/etc/passwd` and the gid values are found in `/etc/group`.

Customarily, Unix systems classify users and groups into two categories: system users and regular users. The system users and groups are given low numbers and regular users and groups have numeric values greater than all the system values. The cutoff for these numbers is found in two parameters in the `/etc/login.defs` configuration file. The default `UID_MIN` value is 1000 and the default `GID_MIN` value is 1000. If a specific uid or gid value is not specified when creating a user with `useradd` or a group with `groupadd` the values assigned will always be above these cutoff values.

Additionally, the [Linux Standard Base](#) recommends that system uid and gid values should be below 100.

Below is a table of suggested uid/gid values used in BLFS beyond those defined in a base LFS installation. These can be changed as desired, but provide a suggested set of consistent values.

Table 3.1. UID/GID Suggested Values

Name	uid	gid
bin	1	
lp	9	
adm		16
atd	17	17
messagebus	18	18
lpadmin		19
named	20	20
gdm	21	21
fcron	22	22
systemd-journal	23	23
apache	25	25
smmsp	26	26
polkitd	27	27
rpc	28	28
exim	31	31
postfix	32	32
postdrop		33
sendmail	34	
mail		34
vmailman	35	35
news	36	36
kdm	37	37
fetchmail	38	
mysql	40	40
postgres	41	41
dovecot	42	42
dovenuill	43	43
ftp	45	45
proftpd	46	46
vsftpd	47	47
rsyncd	48	48

Name	uid	gid
sshd	50	50
stunnel	51	51
dhcpcd	52	52
svn	56	56
svntest		57
git	58	58
games	60	60
kvm		61
wireshark		62
sddm	64	64
lightdm	65	65
scanner		70
colord	71	71
systemd-journal-gateway	73	73
systemd-journal-remote	74	74
systemd-journal-upload	75	75
systemd-network	76	76
systemd-resolve	77	77
systemd-timesync	78	78
systemd-coredump	79	79
uuid	80	80
systemd-oom	81	81
ldap	83	83
avahi	84	84
avahi-autoipd	85	85
netdev		86
ntp	87	87
unbound	88	88
plugdev		90
wheel		97
anonymous	98	
nobody	65534	
nogroup		65534

The Bash Shell Startup Files

The shell program `/bin/bash` (hereafter referred to as just "the shell") uses a collection of startup files to help create an environment. Each file has a specific use and may affect login and interactive environments differently. The files in the `/etc` directory generally provide global settings. If an equivalent file exists in your home directory it may override the global settings.

An interactive login shell is started after a successful login, using `/bin/login`, by reading the `/etc/passwd` file. This shell invocation normally reads `/etc/profile` and its private equivalent `~/.bash_profile` (or `~/.profile` if called as `/bin/sh`) upon startup.

An interactive non-login shell is normally started at the command-line using a shell program (e.g., `[prompt]$ /bin/bash`) or by the `/bin/su` command. An interactive non-login shell is also started with a terminal program such as `xterm` or `kconsole` from within a graphical environment. This type of shell invocation normally copies the parent environment and then reads the user's `~/.bashrc` file for additional startup configuration instructions.

A non-interactive shell is usually present when a shell script is running. It is non-interactive because it is processing a script and not waiting for user input between commands. For these shell invocations, only the environment inherited from the parent shell is used.

The file `~/.bash_logout` is not used for an invocation of the shell. It is read and executed when a user exits from an interactive login shell.

Many distributions use `/etc/bashrc` for system wide initialization of non-login shells. This file is usually called from the user's `~/.bashrc` file and is not built directly into `bash` itself. This convention is followed in this section.

For more information see `info bash -- Nodes: Bash Startup Files and Interactive Shells`.

Note

Most of the instructions below are used to create files located in the `/etc` directory structure which requires you to execute the commands as the `root` user. If you elect to create the files in user's home directories instead, you should run the commands as an unprivileged user.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/bash-shell-startup-files>

/etc/profile

Here is a base `/etc/profile`. This file starts by setting up some helper functions and some basic parameters. It specifies some `bash` history parameters and, for security purposes, disables keeping a permanent history file for the `root` user. It also sets a default user prompt. It then calls small, single purpose scripts in the `/etc/profile.d` directory to provide most of the initialization.

For more information on the escape sequences you can use for your prompt (i.e., the `PS1` environment variable) see `info bash -- Node: Printing a Prompt`.

```
cat > /etc/profile << "EOF"
# Begin /etc/profile
# Written for Beyond Linux From Scratch
# by James Robertson <jameswrobertson@earthlink.net>
# modifications by Dagmar d'Surreal <krivyqntzne@pbzpnfg.arg>

# System wide environment variables and startup programs.

# System wide aliases and functions should go in /etc/bashrc. Personal
# environment variables and startup programs should go into
# ~/.bash_profile. Personal aliases and functions should go into
# ~/.bashrc.

# Functions to help us manage paths. Second argument is the name of the
# path variable to be modified (default: PATH)
pathremove () {
    local IFS=':'
    local NEWPATH
    local DIR
    local PATHVARIABLE=${2:-PATH}
    for DIR in ${!PATHVARIABLE}; do
        if [ "$DIR" != "$1" ]; then
            NEWPATH=$NEWPATH:+$NEWPATH:$DIR
        fi
    done
    export $PATHVARIABLE="$NEWPATH"
}

pathprepend () {
    pathremove $1 $2
    local PATHVARIABLE=${2:-PATH}
    export $PATHVARIABLE="$1${PATHVARIABLE:+:$!PATHVARIABLE}"
}
```

```

pathappend () {
    pathremove $1 $2
    local PATHVARIABLE=${2:-PATH}
    export $PATHVARIABLE="${!PATHVARIABLE:+${!PATHVARIABLE}:}$1"
}

export -f pathremove pathprepend pathappend

# Set the initial path
export PATH=/usr/bin

# Attempt to provide backward compatibility with LFS earlier than 11
if [ ! -L /bin ]; then
    pathappend /bin
fi

if [ $EUID -eq 0 ] ; then
    pathappend /usr/sbin
    if [ ! -L /sbin ]; then
        pathappend /sbin
    fi
    unset HISTFILE
fi

# Set up some environment variables.
export HISTSIZE=1000
export HISTIGNORE="&:[bf]g:exit"

# Set some defaults for graphical systems
export XDG_DATA_DIRS=${XDG_DATA_DIRS:-/usr/share}
export XDG_CONFIG_DIRS=${XDG_CONFIG_DIRS:-/etc/xdg}
export XDG_RUNTIME_DIR=${XDG_RUNTIME_DIR:-/tmp/xdg-$USER}

# Set up a red prompt for root and a green one for users.
NORMAL="\[\e[0m\]"
RED="\[\e[1;31m\]"
GREEN="\[\e[1;32m\]"
if [[ $EUID == 0 ]]; then
    PS1="$RED\u [ $NORMAL\w$RED ]#$ $NORMAL"
else
    PS1="$GREEN\u [ $NORMAL\w$GREEN ]\$ $NORMAL"
fi

for script in /etc/profile.d/*.sh ; do
    if [ -r $script ] ; then
        . $script
    fi
done

unset script RED GREEN NORMAL

# End /etc/profile
EOF

```

The /etc/profile.d Directory

Now create the `/etc/profile.d` directory, where the individual initialization scripts are placed:

```
install --directory --mode=0755 --owner=root --group=root /etc/profile.d
```

/etc/profile.d/bash_completion.sh

Note

Using the bash completion script below is controversial. Not all users like it. It adds many (usually over 1000) lines to the bash environment and makes it difficult to use the 'set' command to examine simple environment variables. Omitting this script does not interfere with the ability of bash to use the tab key for file name completion.

This script imports bash completion scripts, installed by many other BLFS packages, to allow TAB command line completion.

```

cat > /etc/profile.d/bash_completion.sh << "EOF"
# Begin /etc/profile.d/bash_completion.sh
# Import bash completion scripts

# If the bash-completion package is installed, use its configuration instead
if [ -f /usr/share/bash-completion/bash_completion ]; then

    # Check for interactive bash and that we haven't already been sourced.
    if [ -n "${BASH_VERSION-}" -a -n "${PS1-}" -a -z "${BASH_COMPLETION_VERSINFO-}" ]; then

        # Check for recent enough version of bash.
        if [ ${BASH_VERSINFO[0]} -gt 4 ] || \
           [ ${BASH_VERSINFO[0]} -eq 4 -a ${BASH_VERSINFO[1]} -ge 1 ]; then
            [ -r "${XDG_CONFIG_HOME:-$HOME/.config}/bash_completion" ] && \
                . "${XDG_CONFIG_HOME:-$HOME/.config}/bash_completion"
            if shopt -q progcomp && [ -r /usr/share/bash-completion/bash_completion ]; then
                # Source completion code.
                . /usr/share/bash-completion/bash_completion
            fi
        fi
    fi
else

    # bash-completions are not installed, use only bash completion directory
    if shopt -q progcomp; then
        for script in /etc/bash_completion.d/* ; do
            if [ -r $script ] ; then
                . $script
            fi
        done
    fi
fi

# End /etc/profile.d/bash_completion.sh
EOF

```

Make sure that the directory exists:

```
install --directory --mode=0755 --owner=root --group=root /etc/bash_completion.d
```

For a more complete installation, see <https://wiki.linuxfromscratch.org/blfs/wiki/bash-shell-startup-files#bash-completions>.

/etc/profile.d/dircolors.sh

This script uses the `~/.dircolors` and `/etc/dircolors` files to control the colors of file names in a directory listing. They control colorized output of things like `ls --color`. The explanation of how to initialize these files is at the end of this section.

```

cat > /etc/profile.d/dircolors.sh << "EOF"
# Setup for /bin/ls and /bin/grep to support color, the alias is in /etc/bashrc.
if [ -f "/etc/dircolors" ] ; then
    eval $(dircolors -b /etc/dircolors)
fi

if [ -f "$HOME/.dircolors" ] ; then
    eval $(dircolors -b $HOME/.dircolors)
fi

alias ls='ls --color=auto'
alias grep='grep --color=auto'
EOF

```

/etc/profile.d/extrapaths.sh

This script adds some useful paths to the `PATH` and can be used to customize other PATH related environment variables (e.g. `LD_LIBRARY_PATH`, etc) that may be needed for all users.

```

cat > /etc/profile.d/extrapaths.sh << "EOF"
if [ -d /usr/local/lib/pkgconfig ] ; then
    pathappend /usr/local/lib/pkgconfig PKG_CONFIG_PATH
fi
if [ -d /usr/local/bin ] ; then

```

```

        pathprepend /usr/local/bin
fi
if [ -d /usr/local/sbin -a $EUID -eq 0 ]; then
        pathprepend /usr/local/sbin
fi

if [ -d /usr/local/share ]; then
        pathprepend /usr/local/share XDG_DATA_DIRS
fi

# Set some defaults before other applications add to these paths.
pathappend /usr/share/info INFOPATH
EOF

```

Note

The `man` program automatically deduce the search path for man pages by examining the content of the `PATH` variable, see [manpath\(5\)](#) for details. Setting the `MANPATH` variable may override the automatic deduction, so the BLFS editors do not recommend to set it. If you must set it for any reason, it's better to start its value with a colon (:), for example `MANPATH=:/opt/somepkg/share/man:/opt/otherpkg/share/man` so the paths listed in the `MANPATH` variable will be appended to the automatically deduced value instead of overriding it.

/etc/profile.d/readline.sh

This script sets up the default `inputrc` configuration file. If the user does not have individual settings, it uses the global file.

```

cat > /etc/profile.d/readline.sh << "EOF"
# Set up the INPUTRC environment variable.
if [ -z "$INPUTRC" -a ! -f "$HOME/.inputrc" ] ; then
        INPUTRC=/etc/inputrc
fi
export INPUTRC
EOF

```

/etc/profile.d/umask.sh

Setting the `umask` value is important for security. Here the default group write permissions are turned off for system users and when the user name and group name are not the same.

```

cat > /etc/profile.d/umask.sh << "EOF"
# By default, the umask should be set.
if [ "$(id -gn)" = "$(id -un)" -a $EUID -gt 99 ] ; then
        umask 002
else
        umask 022
fi
EOF

```

/etc/profile.d/i18n.sh

This script sets an environment variable necessary for native language support. A full discussion on determining this variable can be found on the [Configuring the System Locale](#) page.

```

cat > /etc/profile.d/i18n.sh << "EOF"
# Set up i18n variables
for i in $(locale); do
        unset ${i%=*}
done

if [[ "$TERM" = linux ]]; then
        export LANG=C.UTF-8
else
        source /etc/locale.conf

        for i in $(locale); do
                key=${i%=*}
                if [[ -v $key ]]; then
                        export $key
                fi
        done
EOF

```

```
fi  
EOF
```

Other Initialization Values

Other initialization can easily be added to the `profile` by adding additional scripts to the `/etc/profile.d` directory.

/etc/bashrc

Here is a base `/etc/bashrc`. Comments in the file should explain everything you need.

```
cat > /etc/bashrc << "EOF"
# Begin /etc/bashrc
# Written for Beyond Linux From Scratch
# by James Robertson <jameswrobertson@earthlink.net>
# updated by Bruce Dubbs <bdubbs@linuxfromscratch.org>

# System wide aliases and functions.

# System wide environment variables and startup programs should go into
# /etc/profile. Personal environment variables and startup programs
# should go into ~/.bash_profile. Personal aliases and functions should
# go into ~/.bashrc

# Provides colored /bin/ls and /bin/grep commands. Used in conjunction
# with code in /etc/profile.

alias ls='ls --color=auto'
alias grep='grep --color=auto'

# Provides prompt for non-login shells, specifically shells started
# in the X environment. [Review the LFS archive thread titled
# PS1 Environment Variable for a great case study behind this script
# addendum.]

NORMAL="\[\e[0m\]"
RED="\[\e[1;31m\]"
GREEN="\[\e[1;32m\]"
if [[ $EUID == 0 ]]; then
    PS1="$RED\u [ $NORMAL\w$RED ]# $NORMAL"
else
    PS1="$GREEN\u [ $NORMAL\w$GREEN ]\$ $NORMAL"
fi

unset RED GREEN NORMAL

# End /etc/bashrc
EOF
```

~/.bash_profile

Here is a base `~/.bash_profile`. If you want each new user to have this file automatically, just change the output of the command to `/etc/skel/.bash_profile` and check the permissions after the command is run. You can then copy `/etc/skel/.bash_profile` to the home directories of already existing users, including `root`, and set the owner and group appropriately.

```
cat > ~/.bash_profile << "EOF"
# Begin ~/.bash_profile
# Written for Beyond Linux From Scratch
# by James Robertson <jameswrobertson@earthlink.net>
# updated by Bruce Dubbs <bdubbs@linuxfromscratch.org>

# Personal environment variables and startup programs.

# Personal aliases and functions should go in ~/.bashrc. System wide
# environment variables and startup programs are in /etc/profile.
# System wide aliases and functions are in /etc/bashrc.

if [ -f "$HOME/.bashrc" ]; then
    source $HOME/.bashrc
fi

if [ -d "$HOME/bin" ]; then
    pathprepend $HOME/bin
```

```

fi

# Having . in the PATH is dangerous
#if [ $EUID -gt 99 ]; then
#  pathappend .
#endif

# End ~/.bash_profile
EOF

```

~/.profile

Here is a base `~/.profile`. The comments and instructions for using `/etc/skel` for `.bash_profile` above also apply here. Only the target file names are different.

```

cat > ~/.profile << "EOF"
# Begin ~/.profile
# Personal environment variables and startup programs.

if [ -d "$HOME/bin" ] ; then
  pathprepend $HOME/bin
fi

# Set up user specific i18n variables
#export LANG=<ll>_<CC>.<charmap><@modifiers>

# End ~/.profile
EOF

```

~/.bashrc

Here is a base `~/.bashrc`.

```

cat > ~/.bashrc << "EOF"
# Begin ~/.bashrc
# Written for Beyond Linux From Scratch
# by James Robertson <jameswrobertson@earthlink.net>

# Personal aliases and functions.

# Personal environment variables and startup programs should go in
# ~/.bash_profile. System wide environment variables and startup
# programs are in /etc/profile. System wide aliases and functions are
# in /etc/bashrc.

if [ -f "/etc/bashrc" ] ; then
  source /etc/bashrc
fi

# Set up user specific i18n variables
#export LANG=<ll>_<CC>.<charmap><@modifiers>

# End ~/.bashrc
EOF

```

~/.bash_logout

This is an empty `~/.bash_logout` that can be used as a template. You will notice that the base `~/.bash_logout` does not include a `clear` command. This is because the `clear` is handled in the `/etc/issue` file.

```

cat > ~/.bash_logout << "EOF"
# Begin ~/.bash_logout
# Written for Beyond Linux From Scratch
# by James Robertson <jameswrobertson@earthlink.net>

# Personal items to perform on logout.

# End ~/.bash_logout
EOF

```

/etc/dircolors

If you want to use the `dircolors` capability, then run the following command. The `/etc/skel` setup steps shown above also can be used here to provide a `~/dircolors` file when a new user is set up. As before, just change the output file name on the following command and assure the permissions, owner, and group are correct on the files created and/or copied.

```
dircolors -p > /etc/dircolors
```

If you wish to customize the colors used for different file types, you can edit the `/etc/dircolors` file. The instructions for setting the colors are embedded in the file.

Finally, Ian Macdonald has written an excellent collection of tips and tricks to enhance your shell environment. You can read it online at <https://www.caliban.org/bash/index.shtml>.

The `/etc/vimrc` and `~/.vimrc` Files

The LFS book installs Vim as its text editor. At this point it should be noted that there are a *lot* of different editing applications out there including Emacs, nano, Joe and many more. Anyone who has been around the Internet (especially usenet) for a short time will certainly have observed at least one flame war, usually involving Vim and Emacs users!

The LFS book creates a basic `vimrc` file. In this section you'll find an attempt to enhance this file. At startup, `vim` reads the global configuration file (`/etc/vimrc`) as well as a user-specific file (`~/.vimrc`). Either or both can be tailored to suit the needs of your particular system.

Here is a slightly expanded `.vimrc` that you can put in `~/.vimrc` to provide user specific effects. Of course, if you put it into `/etc/skel/.vimrc` instead, it will be made available to users you add to the system later. You can also copy the file from `/etc/skel/.vimrc` to the home directory of users already on the system, such as `root`. Be sure to set permissions, owner, and group if you do copy anything directly from `/etc/skel`.

```
" Begin .vimrc

set columns=80
set wrapmargin=8
set ruler

" End .vimrc
```

Note that the comment tags are " instead of the more usual # or //. This is correct, the syntax for `vimrc` is slightly unusual.

Below you'll find a quick explanation of what each of the options in this example file means here:

- `set columns=80`: This simply sets the number of columns used on the screen.
- `set wrapmargin=8`: This is the number of characters from the right window border where wrapping starts.
- `set ruler`: This makes `vim` show the current row and column at the bottom right of the screen.

More information on the *many* `vim` options can be found by reading the help inside `vim` itself. Do this by typing `:help` in `vim` to get the general help, or by typing `:help usr_toc.txt` to view the User Manual Table of Contents.

Customizing your Logon with `/etc/issue`

When you first boot up your new LFS system, the logon screen will be nice and plain (as it should be in a bare-bones system). Many people however, will want their system to display some information in the logon message. This can be accomplished using the file `/etc/issue`.

The `/etc/issue` file is a plain text file which will also accept certain escape sequences (see below) in order to insert information about the system. There is also the file `issue.net` which can be used when logging on remotely. `ssh` however, will only use it if you set the option in the configuration file and will *not* interpret the escape sequences shown below.

One of the most common things which people want to do is clear the screen at each logon. The easiest way of doing that is to put a "clear" escape sequence into `/etc/issue`. A simple way of doing this is to issue the command `clear > /etc/issue`. This will insert the relevant escape code into the start of the `/etc/issue` file. Note that if you do this, when you edit the file, you should leave the characters (normally '^[[H^[[2J') on the first line alone.

Note

Terminal escape sequences are special codes recognized by the terminal. The ^[represents an ASCII ESC character. The sequence ESC [H puts the cursor in the upper left hand corner of the screen and ESC 2 J

erases the screen. For more information on terminal escape sequences see <https://invisible-mirror.net/xterm/ctlseqs/ctlseqs.html>

The following sequences are recognized by `agetty` (the program which usually parses `/etc/issue`). This information is from `man agetty` where you can find extra information about the logon process.

The `issue` file can contain certain character sequences to display various information. All `issue` sequences consist of a backslash (\) immediately followed by one of the letters explained below (so \d in `/etc/issue` would insert the current date).

- b Insert the baudrate of the current line.
- d Insert the current date.
- s Insert the system name, the name of the operating system.
- l Insert the name of the current tty line.
- m Insert the architecture identifier of the machine, e.g., i686.
- n Insert the nodename of the machine, also known as the hostname.
- o Insert the domainname of the machine.
- r Insert the release number of the kernel, e.g., 2.6.11.12.
- t Insert the current time.
- u Insert the number of current users logged in.
- U Insert the string "1 user" or "<n> users" where <n> is the number of current users logged in.
- v Insert the version of the OS, e.g., the build-date etc.

Chapter 4. Security

Security takes many forms in a computing environment. After some initial discussion, this chapter gives examples of three different types of security: access, prevention and detection.

Access for users is usually handled by `login` or an application designed to handle the login function. In this chapter, we show how to enhance `login` by setting policies with PAM modules. Access via networks can also be secured by policies set by `iptables`, commonly referred to as a firewall. The Network Security Services (NSS) and Netscape Portable Runtime (NSPR) libraries can be installed and shared among the many applications requiring them. For applications that don't offer the best security, you can use the Stunnel package to wrap an application daemon inside an SSL tunnel.

Prevention of breaches, like a trojan, are assisted by applications like GnuPG, specifically the ability to confirm signed packages, which recognizes modifications of the tarball after the packager creates it.

Finally, we touch on detection with a package that stores "signatures" of critical files (defined by the administrator) and then regenerates those "signatures" and compares for files that have been changed.

Vulnerabilities

About vulnerabilities

All software has bugs. Sometimes, a bug can be exploited, for example to allow users to gain enhanced privileges (perhaps gaining a root shell, or simply accessing or deleting other user's files), or to allow a remote site to crash an application (denial of service), or for theft of data. These bugs are labelled as vulnerabilities.

The main place where vulnerabilities get logged is cve.mitre.org. Unfortunately, many vulnerability numbers (CVE-yyyy-nnnn) are initially only labelled as "reserved" when distributions start issuing fixes. Also, some vulnerabilities apply to particular combinations of `configure` options, or only apply to old versions of packages which have long since been updated in BLFS.

BLFS differs from distributions—there is no BLFS security team, and the editors only become aware of vulnerabilities after they are public knowledge. Sometimes, a package with a vulnerability will not be updated in the book for a long time. Issues can be logged in the Trac system, which might speed up resolution.

The normal way for BLFS to fix a vulnerability is, ideally, to update the book to a new fixed release of the package. Sometimes that happens even before the vulnerability is public knowledge, so there is no guarantee that it will be shown as a vulnerability fix in the Changelog. Alternatively, a `sed` command, or a patch taken from a distribution, may be appropriate.

The bottom line is that you are responsible for your own security, and for assessing the potential impact of any problems.

The editors now issue Security Advisories for packages in BLFS (and LFS), which can be found at [BLFS Security Advisories](#), and grade the severity according to what upstream reports, or to what is shown at nvd.nist.gov if that has details.

To keep track of what is being discovered, you may wish to follow the security announcements of one or more distributions. For example, Debian has [Debian security](#). Fedora's links on security are at [the Fedora wiki](#). Details of Gentoo linux security announcements are discussed at [Gentoo security](#). Finally, the Slackware archives of security announcements are at [Slackware security](#).

The most general English source is perhaps [the Full Disclosure Mailing List](#), but please read the comment on that page. If you use other languages you may prefer other sites such as [heise.de](#) (German) or [cert.hr](#) (Croatian). These are not linux-specific. There is also a daily update at lwn.net for subscribers (free access to the data after 2 weeks, but their vulnerabilities database at [lwn.net/Alerts](#) is unrestricted).

For some packages, subscribing to their 'announce' lists will provide prompt news of newer versions.

make-ca-1.14

Introduction to make-ca

Public Key Infrastructure (PKI) is a method to validate the authenticity of an otherwise unknown entity across untrusted networks. PKI works by establishing a chain of trust, rather than trusting each individual host or entity explicitly. In order for a certificate presented by a remote entity to be trusted, that certificate must present a complete chain of certificates that can be validated using the root certificate of a Certificate Authority (CA) that is trusted by the local machine.

Establishing trust with a CA involves validating things like company address, ownership, contact information, etc., and ensuring that the CA has followed best practices, such as undergoing periodic security audits by independent investigators and maintaining an always available certificate revocation list. This is well outside the scope of BLFS (as it is for most Linux distributions). The certificate store provided here is taken from the Mozilla Foundation, who have established very strict inclusion policies described [here](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lfs-book/make-ca/archive/v1.14/make-ca-1.14.tar.gz>
- Download size: 40 KB
- Download MD5 Sum: e99d2985ead0037caedb765fd66b33f0
- Estimated disk space required: 164 KB (with all runtime deps)
- Estimated build time: 0.1 SBU (with all runtime deps)

Note

This package ships a CA certificate for validating the identity of <https://hg.mozilla.org/>. If the trust chain of this website has been changed after the release of make-ca-1.14, it may fail to get the revision of certdata.txt from server. Use an updated make-ca release at [the release page](#) if this issue happens.

make-ca Dependencies

Required

[p11-kit-0.25.5](#) (runtime, built after [libtasn1-4.19.0](#), required in the following instructions to generate certificate stores from trust anchors, and each time make-ca is run)

Optional (runtime)

[nss-3.103](#) (to generate a shared NSSDB)

Installation of make-ca and Generation of the CA-certificates stores

The make-ca script will download and process the certificates included in the certdata.txt file for use as trust anchors for the [p11-kit-0.25.5](#) trust module. Additionally, it will generate system certificate stores used by BLFS applications (if the recommended and optional applications are present on the system). Any local certificates stored in /etc/ssl/local will be imported to both the trust anchors and the generated certificate stores (overriding Mozilla's trust). Additionally, any modified trust values will be copied from the trust anchors to /etc/ssl/local prior to any updates, preserving custom trust values that differ from Mozilla when using the trust utility from p11-kit to operate on the trust store.

To install the various certificate stores, first install the make-ca script into the correct location. As the `root` user:

```
make install &&
install -vdm755 /etc/ssl/local
```

Note

Technically, this package is already installed at this point. But most packages listing make-ca as a dependency actually require the system certificate store set up by this package, rather than the `make-ca` program itself. So the instructions for using `make-ca` for setting up the system certificate store are included in this section. You should make sure the required runtime dependency for `make-ca` is satisfied now, and continue to follow the instructions.

As the `root` user, download the certificate source and prepare for system use with the following command:

Note

If running the script a second time with the same version of `certdata.txt`, for instance, to update the stores when `make-ca` is upgraded, or to add additional stores as the requisite software is installed, replace the `-g` switch with the `-r` switch in the command line. If packaging, run `make-ca --help` to see all available command line options.

```
/usr/sbin/make-ca -g
```

You should periodically update the store with the above command, either manually, or via a `systemd` timer. A timer is installed at `/usr/lib/systemd/system/update-pki.timer` that, if enabled, will check for updates weekly. Execute the following commands, as the `root` user, to enable the `systemd` timer:

```
systemctl enable update-pki.timer
```

Configuring make-ca

For most users, no additional configuration is necessary, however, the default `certdata.txt` file provided by `make-ca` is obtained from the `mozilla-release` branch, and is modified to provide a Mercurial revision. This will be the correct version for most systems. There are several other variants of the file available for use that might be preferred for one reason or another, including the files shipped with Mozilla products in this book. RedHat and OpenSUSE, for instance, use the version included in [nss-3.103](#). Additional upstream downloads are available at the links included in `/etc/make-ca/make-ca.conf.dist`. Simply copy the file to `/etc/make-ca.conf` and edit as appropriate.

About Trust Arguments

There are three trust types that are recognized by the `make-ca` script, SSL/TLS, S/Mime, and code signing. For OpenSSL, these are `serverAuth`, `emailProtection`, and `codeSigning` respectively. If one of the three trust arguments is omitted, the certificate is neither trusted, nor rejected for that role. Clients that use OpenSSL or NSS encountering this certificate will present a warning to the user. Clients using GnuTLS without p11-kit support are not aware of trusted certificates. To include this CA into the `ca-bundle.crt`, `email-ca-bundle.crt`, or `objsign-ca-bundle.crt` files (the GnuTLS legacy bundles), it must have the appropriate trust arguments.

Adding Additional CA Certificates

The `/etc/ssl/local` directory is available to add additional CA certificates to the system trust store. This directory is also used to store certificates that were added to or modified in the system trust store by [p11-kit-0.25.5](#) so that trust values are maintained across upgrades. Files in this directory must be in the OpenSSL trusted certificate format. Certificates imported using the `trust` utility from [p11-kit-0.25.5](#) will utilize the x509 Extended Key Usage values to assign default trust values for the system anchors.

If you need to override trust values, or otherwise need to create an OpenSSL trusted certificate manually from a regular PEM encoded file, you need to add trust arguments to the `openssl` command, and create a new certificate. For example, using the [CAcert](#) roots, if you want to trust both for all three roles, the following commands will create appropriate OpenSSL trusted certificates (run as the `root` user after [Wget-1.24.5](#) is installed):

```
wget http://www.cacert.org/certs/root.crt &&
wget http://www.cacert.org/certs/class3.crt &&
openssl x509 -in root.crt -text -fingerprint -setalias "CAcert Class 1 root" \
    -addtrust serverAuth -addtrust emailProtection -addtrust codeSigning \
    > /etc/ssl/local/CAcert_Class_1_root.pem &&
openssl x509 -in class3.crt -text -fingerprint -setalias "CAcert Class 3 root" \
    -addtrust serverAuth -addtrust emailProtection -addtrust codeSigning \
    > /etc/ssl/local/CAcert_Class_3_root.pem &&
/usr/sbin/make-ca -r
```

Overriding Mozilla Trust

Occasionally, there may be instances where you don't agree with Mozilla's inclusion of a particular certificate authority. If you'd like to override the default trust of a particular CA, simply create a copy of the existing certificate in `/etc/ssl/local` with different trust arguments. For example, if you'd like to distrust the "Makebelieve_CA_Root" file, run the following commands:

```
openssl x509 -in /etc/ssl/certs/Makebelieve_CA_Root.pem \
  -text \
  -fingerprint \
  -setalias "Disabled Makebelieve CA Root" \
  -addreject serverAuth \
  -addreject emailProtection \
  -addreject codeSigning \
> /etc/ssl/local/Disabled_Makebelieve_CA_Root.pem &&
/usr/sbin/make-ca -r
```

Using make-ca with Python3

When Python3 was installed in LFS, it included the `pip3` module with vendored certificates from the Certifi module. That was necessary, but it means that whenever `pip3` is used it can reference those certificates, primarily when creating a virtual environment or when installing a module with all its wheel dependencies in one go.

It is generally considered that the System Administrator should be in charge of which certificates are available. Now that [make-ca-1.14](#) and [p11-kit-0.25.5](#) have been installed and `make-ca` has been configured, it is possible to make `pip3` use the system certificates.

The vendored certificates installed in LFS are a snapshot from when the pulled-in version of Certifi was created. If you regularly update the system certificates, the vendored version will become out of date.

To use the system certificates in Python3, you should set `_PIP_STANDALONE_CERT` to point to them, e.g for the bash shell:

```
export _PIP_STANDALONE_CERT=/etc/pki/tls/certs/ca-bundle.crt
```

Warning

If you have created virtual environments, for example when testing modules, and those include the Requests and Certifi modules in `~/.local/lib/python3.12/`, then those local modules will be used instead of the system certificates unless you remove the local modules.

To use the system certificates in Python3 with the BLFS profiles, add the following variable to your system or personal profiles:

```
mkdir -pv /etc/profile.d &&
cat > /etc/profile.d/pythoncerts.sh << "EOF"
# Begin /etc/profile.d/pythoncerts.sh

export _PIP_STANDALONE_CERT=/etc/pki/tls/certs/ca-bundle.crt

# End /etc/profile.d/pythoncerts.sh
EOF
```

Contents

Installed Programs: make-ca

Installed Directories: /etc/ssl/{certs,local} and /etc/pki/{nssdb,anchors,tls}/{certs,java}

Short Descriptions

`make-ca` is a shell script that adapts a current version of `certdata.txt`, and prepares it for use as the system trust store

CrackLib-2.10.2

Introduction to CrackLib

The CrackLib package contains a library used to enforce strong passwords by comparing user selected passwords to words in chosen word lists.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/cracklib/cracklib/releases/download/v2.10.2/cracklib-2.10.2.tar.xz>
- Download MD5 sum: a99e0aef4c677df7063624690b634988
- Download size: 456 KB
- Estimated disk space required: 5.0 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

Recommended word list for English-speaking countries:

- Download (HTTP): <https://github.com/cracklib/cracklib/releases/download/v2.10.2/cracklib-words-2.10.2.xz>
- Download MD5 sum: f27804022dbf2682a7f7c353317f9a53
- Download size: 4.0 MB

There are additional word lists available for download, e.g., from <https://wiki.skullsecurity.org/index.php/Passwords>. CrackLib can utilize as many, or as few word lists you choose to install.

Important

Users tend to base their passwords on regular words of the spoken language, and crackers know that. CrackLib is intended to filter out such bad passwords at the source using a dictionary created from word lists. To accomplish this, the word list(s) for use with CrackLib must be an exhaustive list of words and word-based keystroke combinations likely to be chosen by users of the system as (guessable) passwords.

The default word list recommended above for downloading mostly satisfies this role in English-speaking countries. In other situations, it may be necessary to download (or even create) additional word lists.

Note that word lists suitable for spell-checking are not usable as CrackLib word lists in countries with non-Latin based alphabets, because of “word-based keystroke combinations” that make bad passwords.

Installation of CrackLib

Install CrackLib by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --with-default-dict=/usr/lib/cracklib/pw_dict &&
make
```

Now, as the `root` user:

```
make install
```

Issue the following commands as the `root` user to install the recommended word list and create the CrackLib dictionary. Other word lists (text based, one word per line) can also be used by simply installing them into `/usr/share/dict` and adding them to the `create-cracklib-dict` command.

```
install -v -m644 -D    ./cracklib-words-2.10.2.xz \
                        /usr/share/dict/cracklib-words.xz    &&
unxz -v                /usr/share/dict/cracklib-words.xz    &&
ln -v -sf cracklib-words /usr/share/dict/words           &&
echo $(hostname) >>   /usr/share/dict/cracklib-extra-words &&
install -v -m755 -d     /usr/lib/cracklib               &&
create-cracklib-dict   /usr/share/dict/cracklib-words \
                        /usr/share/dict/cracklib-extra-words
```

If desired, check the proper operation of the library as an unprivileged user by issuing the following command:

```
make test
```

Important

If you are installing CrackLib after your LFS system has been completed and you have the Shadow package installed, you must reinstall [Shadow-4.16.0](#) if you wish to provide strong password support on your system. If you are now going to install the [Linux-PAM-1.6.1](#) package, you may disregard this note as Shadow will be reinstalled after the Linux-PAM installation.

Command Explanations

`--with-default-dict=/usr/lib/cracklib/pw_dict`: This parameter forces the installation of the CrackLib dictionary to the `/lib` hierarchy.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`install -v -m644 -D ...`: This command creates the `/usr/share/dict` directory (if it doesn't already exist) and installs the compressed word list there.

`ln -v -s cracklib-words /usr/share/dict/words`: The word list is linked to `/usr/share/dict/words` as historically, `words` is the primary word list in the `/usr/share/dict` directory. Omit this command if you already have a `/usr/share/dict/words` file installed on your system.

`echo $(hostname) >>...`: The value of `hostname` is echoed to a file called `cracklib-extra-words`. This extra file is intended to be a site specific list which includes easy to guess passwords such as company or department names, user names, product names, computer names, domain names, etc.

`create-cracklib-dict ...`: This command creates the CrackLib dictionary from the word lists. Modify the command to add any additional word lists you have installed.

Contents

Installed Programs: cracklib-check, cracklib-format, cracklib-packer, cracklib-unpacker, cracklib-update, and create-cracklib-dict

Installed Libraries: libcrack.so and the _cracklib.so (Python module)

Installed Directories: /usr/lib/cracklib, /usr/share/dict and /usr/share/cracklib

Short Descriptions

<code>cracklib-check</code>	is used to determine if a password is strong
<code>cracklib-format</code>	is used to format text files (lowercases all words, removes control characters and sorts the lists)
<code>cracklib-packer</code>	creates a database with words read from standard input
<code>cracklib-unpacker</code>	displays on standard output the database specified
<code>create-cracklib-dict</code>	is used to create the CrackLib dictionary from the given word list(s)
<code>libcrack.so</code>	provides a fast dictionary lookup method for strong password enforcement

cryptsetup-2.7.4

Introduction to cryptsetup

cryptsetup is used to set up transparent encryption of block devices using the kernel crypto API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/utils/cryptsetup/v2.7/cryptsetup-2.7.4.tar.xz>
- Download MD5 sum: 26ffe48f65d144af91b2a9639425d08c
- Download size: 11 MB
- Estimated disk space required: 35 MB (add 5 MB for tests)
- Estimated build time: 0.2 SBU (add 8.1 SBU for tests)

cryptsetup Dependencies

Required

[JSON-C-0.17](#), [LVM2-2.03.26](#), and [popt-1.19](#)

Optional

[asciidoc-2.0.23](#), [libpwquality-1.4.5](#), [argon2](#), [libssh](#), and [passwdqc](#)

Kernel Configuration

Encrypted block devices require kernel support. To use it, the appropriate kernel configuration parameters need to be set:

```

Device Drivers --->
[*] Multiple devices driver support (RAID and LVM) ---> [MD]
  <*/M> Device mapper support [BLK_DEV_DM]
  <*/M> Crypt target support [DM_CRYPT]

--- Cryptographic API ---> [CRYPTO]
  Block ciphers --->
    <*/M> AES (Advanced Encryption Standard) [CRYPTO_AES]
    # For tests:
    <*/M> Twofish [CRYPTO_TWOFISH]
  Length-preserving ciphers and modes --->
    <*/M> XTS (XOR Encrypt XOR with ciphertext stealing) [CRYPTO_XTS]
  Hashes, digests, and MACs --->
    <*/M> SHA-224 and SHA-256 [CRYPTO_SHA256]
  Userspace interface --->
    <*/M> Symmetric key cipher algorithms [CRYPTO_USER_API_SKCIPHER]

```

Installation of cryptsetup

Install cryptsetup by running the following commands:

```

./configure --prefix=/usr      \
--disable-ssh-token \
--disable-asciidoc  &&
make

```

To test the result, issue as the `root` user: `make check`. Some tests will fail if appropriate kernel configuration options are not set. Some additional options that may be needed for tests are:

```

CONFIG_SCSI_LOWLEVEL,
CONFIG_SCSI_DEBUG,
CONFIG_BLK_DEV_DM_BUILTIN,
CONFIG_CRYPTO_USER,
CONFIG_CRYPTO_CRYPTD,
CONFIG_CRYPTO_LRW,
CONFIG_CRYPTO_XTS,
CONFIG_CRYPTO_ESSIV,
CONFIG_CRYPTO_CRC10DIF,
CONFIG_CRYPTO_AES_TI,
CONFIG_CRYPTO_AES_NI_INTEL,
CONFIG_CRYPTO_BLOWFISH,
CONFIG_CRYPTO_CAST5,
CONFIG_CRYPTO_SERPENT,
CONFIG_CRYPTO_SERPENT_SSE2_X86_64,
CONFIG_CRYPTO_SERPENT_AVX_X86_64,
CONFIG_CRYPTO_SERPENT_AVX2_X86_64, and
CONFIG_CRYPTO_TWOFISH_X86_64

```

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-ssh-token`: This switch is required if the optional libssh dependency is not installed.

--disable-asciidoc: This switch disables regeneration of the man pages. Remove this switch if you have [asciidoc-2.0.23](#) installed and wish to regenerate the man pages. Note that even if this switch is used, the pre-generated man pages are shipped in the tarball and they'll still be installed.

Configuring cryptsetup

Because of the number of possible configurations, setup of encrypted volumes is beyond the scope of the BLFS book. Please see the configuration guide in the cryptsetup [FAQ](#).

Contents

Installed Programs: cryptsetup, cryptsetup-reencrypt, integritysetup, and veritysetup

Installed Libraries: libcryptsetup.so

Installed Directories: None

Short Descriptions

<code>cryptsetup</code>	is used to setup dm-crypt managed device-mapper mappings
<code>cryptsetup-</code> <code>reencrypt</code>	is a tool for offline LUKS device re-encryption
<code>integritysetup</code>	is a tool to manage dm-integrity (block level integrity) volumes
<code>veritysetup</code>	is used to configure dm-verity managed device-mapper mappings. The Device-mapper verity target provides read-only transparent integrity checking of block devices using the kernel crypto API

Cyrus SASL-2.1.28

Introduction to Cyrus SASL

The Cyrus SASL package contains a Simple Authentication and Security Layer implementation, a method for adding authentication support to connection-based protocols. To use SASL, a protocol includes a command for identifying and authenticating a user to a server and for optionally negotiating protection of subsequent protocol interactions. If its use is negotiated, a security layer is inserted between the protocol and the connection.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/cyrusimap/cyrus-sasl/releases/download/cyrus-sasl-2.1.28/cyrus-sasl-2.1.28.tar.gz>
- Download MD5 sum: 6f228a692516f5318a64505b46966cfa
- Download size: 3.9 MB
- Estimated disk space required: 28 MB
- Estimated build time: 0.2 SBU

Cyrus SASL Dependencies

Recommended

[Imdb-0.9.31](#)

Optional

[Linux-PAM-1.6.1](#), [MIT Kerberos V5-1.21.3](#), [MariaDB-10.11.8](#) or [MySQL](#), [OpenLDAP-2.6.8](#), [PostgreSQL-16.4](#), [sphinx-8.0.2](#), [SQLite-3.46.1](#), [Berkeley DB](#) (deprecated), [krb4](#), [Dmalloc](#), and [Pod::POM::View::Restructured](#)

Installation of Cyrus SASL

Note

This package does not support parallel build.

First, fix a problem revealed by gcc-14:

```
sed '/saslnt/a #include <time.h>' -i lib/saslutil.c &&  
sed '/plugin_common/a #include <time.h>' -i plugins/cram.c
```

Install Cyrus SASL by running the following commands:

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --enable-auth-sasldb \
            --with-dblib=lmdb \
            --with-dbpath=/var/lib/sasl/sasldb2 \
            --with-sphinx-build=no \
            --with-saslauthd=/var/run/saslauthd &&
make -j1
```

This package does not come with a test suite. If you are planning on using the GSSAPI authentication mechanism, test it after installing the package using the sample server and client programs which were built in the preceding step.

Instructions for performing the tests can be found at

<https://www.linuxfromscratch.org/hints/downloads/files/cyrus-sasl.txt>

Now, as the *root* user:

```
make install &&
install -v -dm755                               /usr/share/doc/cyrus-sasl-2.1.28/html &&
install -v -m644  saslauthd/LDAP_SASLAUTHD /usr/share/doc/cyrus-sasl-2.1.28      &&
install -v -m644  doc/legacy/*.html       /usr/share/doc/cyrus-sasl-2.1.28/html &&
install -v -dm700 /var/lib/sasl
```

Command Explanations

--with-dbpath=/var/lib/sasl/sasldb2: This switch forces the sasldb database to be created in /var/lib/sasl instead of /etc.

--with-saslauthd=/var/run/saslauthd: This switch forces `saslauthd` to use the FHS compliant directory `/var/run/saslauthd` for variable run-time data.

--enable-auth-saslbd: This switch enables SASLDB authentication backend.

--with-dblib=gdbm: This switch forces GDBM to be used instead of LMDB.

--with-ldap: This switch enables the OpenLDAP support.

--enable-ldapdb: This switch enables the LDAPDB authentication backend.

--enable-login: This option enables unsupported LOGIN authentication.

--enable-ntlm: This option enables unsupported NTLM authentication.

`install -v -m644 ...`: These commands install documentation which is

`install -v -m700 -d /var/lib/sasl`: This directory must exist when starting `saslauthd` or using the `sasldb` plugin. If you

going to be running the daemon or using the plugins, you may omit the creation of this directory.

• 8 •

defined name of the application)

Configuration Information

[See `http://www.apache.org/licenses/LICENSE-2.0.txt`](#) for information on what to include in the application configuration files.

See [http://www.cs.cmu.edu/~sas/21203-12110/CHANGES](#) for configuring basildand with OPENBLAS.

See <https://www.cyfrusimap.org/sasi/sasi/gssapi.html#gssapi> for configuring saslauthd with Kerberos.

System Unit

If you need to run the `saslauthd` daemon at system startup, install the `saslauthd.service` unit included in the [blfs-systemd-units-20240801](#) package using the following command:

```
make install-saslauthd
```

Note

You'll need to modify `/etc/default/saslauthd` and modify the `MECHANISM` parameter with your desired authentication mechanism. The default authentication mechanism is "shadow".

Contents

Installed Programs: pluginviewer, saslauthd, sasldblistusers2, saslpasswd2, and testsaslauthd

Installed Library: libsasl2.so

Installed Directories: /usr/include/sasl, /usr/lib/sasl2, /usr/share/doc/cyrus-sasl-2.1.28 and /var/lib/sasl

Short Descriptions

<code>pluginviewer</code>	is used to list loadable SASL plugins and their properties
<code>saslauthd</code>	is the SASL authentication server
<code>sasldblistusers2</code>	is used to list the users in the SASL password database <code>sasldb2</code>
<code>saslpasswd2</code>	is used to set and delete a user's SASL password and mechanism specific secrets in the SASL password database <code>sasldb2</code>
<code>testsaslauthd</code>	is a test utility for the SASL authentication server
<code>libsasl2.so</code>	is a general purpose authentication library for server and client applications

GnuPG-2.4.5

Introduction to GnuPG

The GnuPG package is GNU's tool for secure communication and data storage. It can be used to encrypt data and to create digital signatures. It includes an advanced key management facility and is compliant with the proposed OpenPGP Internet standard as described in RFC2440 and the S/MIME standard as described by several RFCs. GnuPG 2 is the stable version of GnuPG integrating support for OpenPGP and S/MIME.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/gnupg/gnupg-2.4.5.tar.bz2>
- Download MD5 sum: 49c3534e87744e994250d37c1b43f928
- Download size: 7.6 MB
- Estimated disk space required: 164 MB (with tests)
- Estimated build time: 1.2 SBU (with tests; using parallelism=4)

GnuPG 2 Dependencies

Required

[libassuan-3.0.1](#), [libgcrypt-1.11.0](#), [libksba-1.6.7](#), [npth-1.7](#), and [OpenLDAP-2.6.8](#)

Recommended

[GnuTLS-3.8.7.1](#) (required to communicate with keyservers using https or hkps protocol) and [pinentry-1.3.1](#) (Run-time requirement for most of the package's functionality)

Optional

[curl-8.9.1](#), [fuse-3.16.2](#), [ImageMagick-7.1.1-36](#) (for the `convert` utility, used for generating the documentation), [ibus-1.0.27](#), an [MTA](#), [SQLite-3.46.1](#), [texlive-20240312](#) (or [install-tl-unx](#)), [fig2dev](#) (for generating documentation), and [GNU adns](#)

Installation of GnuPG

Install GnuPG by running the following commands:

```
mkdir build &&
cd build &&

./configure --prefix=/usr          \
            --localstatedir=/var \
            --sysconfdir=/etc   \
            --docdir=/usr/share/doc/gnupg-2.4.5 &&
make &&

makeinfo --html --no-split -I doc -o doc/gnupg_nochunks.html ../doc/gnupg.texi &&
makeinfo --plaintext      -I doc -o doc/gnupg.txt           ../doc/gnupg.texi &&
make -C doc html
```

If you have [texlive-20240312](#) installed and you wish to create documentation in the pdf format, issue the following command:

```
make -C doc pdf
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&

install -v -m755 -d /usr/share/doc/gnupg-2.4.5/html          &&
install -v -m644   doc/gnupg_nochunks.html \
                  /usr/share/doc/gnupg-2.4.5/html/gnupg.html &&
install -v -m644   ../doc/*.texi doc/gnupg.txt \
                  /usr/share/doc/gnupg-2.4.5 &&
install -v -m644   doc/gnupg.html/* \
                  /usr/share/doc/gnupg-2.4.5/html
```

If you created the pdf format of the documentation, install them using the following command as the `root` user:

```
install -v -m644 doc/gnupg.pdf \
                  /usr/share/doc/gnupg-2.4.5
```

Command Explanations

`mkdir build && cd build`: the Gnupg2 developers recommend to build the package in a dedicated directory.

`--docdir=/usr/share/doc/gnupg-2.4.5`: This switch changes the default docdir to `/usr/share/doc/gnupg-2.4.5`.

`--enable-all-tests`: This switch allows more tests to be run with `make check`.

`--enable-g13`: This switch enables building the g13 program.

Contents

Installed Programs: addgnupghome, applygnupgdefaults, dirmngr, dirmngr-client, g13 (optional), gpg-agent, gpg-card, gpg-connect-agent, gpg, gpgconf, gpparsemail, gpgsm, gpgsm, gpgsplit, gptar, gpgv, gpg-wks-client, gpg-wks-server, kbxutil, and watchgnupg

Installed Libraries: None

Installed Directories: /usr/share/doc/gnupg-2.4.5 and /usr/share/gnupg

Short Descriptions

<code>addgnupghome</code>	is used to create and populate a user's <code>~/.gnupg</code> directories
<code>applygnupgdefaults</code>	is a wrapper script used to run <code>gpgconf</code> with the <code>--apply-defaults</code> parameter on all user's GnuPG home directories
<code>dirmngr</code>	is a tool that takes care of accessing the OpenPGP keyservers
<code>dirmngr-client</code>	is a tool to contact a running dirmngr and test whether a certificate has been revoked
<code>g13</code>	is a tool to create, mount or unmount an encrypted file system container (optional)

<code>gpg-agent</code>	is a daemon used to manage secret (private) keys independently from any protocol. It is used as a backend for <code>gpg</code> and <code>gpgsm</code> as well as for a couple of other utilities
<code>gpg-card</code>	is a tool to manage smart cards and tokens
<code>gpg-connect-agent</code>	is a utility used to communicate with a running <code>gpg-agent</code>
<code>gpg</code>	is the OpenPGP part of the GNU Privacy Guard (GnuPG). It is a tool used to provide digital encryption and signing services using the OpenPGP standard
<code>gpgconf</code>	is a utility used to automatically and reasonably safely query and modify configuration files in the <code>~/.gnupg</code> home directory. It is designed not to be invoked manually by the user, but automatically by graphical user interfaces
<code>gpgparsemail</code>	is a utility currently only useful for debugging. Run it with <code>--help</code> for usage information
<code>gpgscm</code>	executes the given scheme program or spawns an interactive shell
<code>gpgsm</code>	is a tool similar to <code>gpg</code> used to provide digital encryption and signing services on X.509 certificates and the CMS protocol. It is mainly used as a backend for S/MIME mail processing
<code>gpgsplit</code>	splits an OpenPGP message into packets
<code>gpgtar</code>	is a tool to encrypt or sign files into an archive
<code>gpgv</code>	is a verify only version of <code>gpg</code>
<code>gpg-wks-client</code>	is a client for the Web Key Service protocol
<code>gpg-wks-server</code>	provides a server for the Web Key Service protocol
<code>kbxutil</code>	is used to list, export and import Keybox data
<code>watchgnupg</code>	is used to listen to a Unix Domain socket created by any of the GnuPG tools

GnuTLS-3.8.7.1

Introduction to GnuTLS

The GnuTLS package contains libraries and userspace tools which provide a secure layer over a reliable transport layer. Currently the GnuTLS library implements the proposed standards by the IETF's TLS working group. Quoting from the [TLS 1.3 protocol specification](#) :

“ TLS allows client/server applications to communicate over the Internet in a way that is designed to prevent eavesdropping, tampering, and message forgery. ”

GnuTLS provides support for TLS 1.3, TLS 1.2, TLS 1.1, TLS 1.0, and (optionally) SSL 3.0 protocols. It also supports TLS extensions, including server name and max record size. Additionally, the library supports authentication using the SRP protocol, X.509 certificates, and OpenPGP keys, along with support for the TLS Pre-Shared-Keys (PSK) extension, the Inner Application (TLS/IA) extension, and X.509 and OpenPGP certificate handling.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/gnutls/v3.8/gnutls-3.8.7.1.tar.xz>
- Download MD5 sum: 46777802233eb7203d47a8b8880c8c90
- Download size: 6.4 MB
- Estimated disk space required: 154 MB (add 115 MB for tests)
- Estimated build time: 0.5 SBU (add 1.5 SBU for tests; both using parallelism=8)

Note

When extracting this package tarball, it expands to the gnutls-3.8.7 directory, instead of the expected gnutls-3.8.7.1 directory.

GnuTLS Dependencies

Required

[Nettle-3.10](#)

Recommended

[make-ca-1.14](#), [libunistring-1.2](#), [libtasn1-4.19.0](#), and [p11-kit-0.25.5](#)

Optional

[Brotli-1.1.0](#), [Doxygen-1.12.0](#), [GTK-Doc-1.34.0](#), [libidn-1.42](#) or [libidn2-2.3.7](#), [libseccomp-2.5.5](#), [Net-tools-2.10](#) (used during the test suite), [texlive-20240312](#) or [install-tl-unx](#), [Unbound-1.21.0](#) (to build the DANE library), [Valgrind-3.23.0](#) (used during the test suite), [autogen](#), [cmocka](#) and [datefudge](#) (used during the test suite if the DANE library is built), and [Trousers](#) (Trusted Platform Module support)

Note

Note that if you do not install [libtasn1-4.19.0](#), a version shipped in the GnuTLS tarball will be used instead.

Installation of GnuTLS

Install GnuTLS by running the following commands:

```
./configure --prefix=/usr \
            --docdir=/usr/share/doc/gnutls-3.8.7.1 \
            --with-default-trust-store-pkcs11="pkcs11:" &&
make
```

One test hangs the test procedure. Disable it: `sed '/ocsp-must-staple-connection/d' -i tests/Makefile`. To test the results, now issue: `make check`.

Now, install the package as the `root` user:

```
make install
```

Command Explanations

`--with-default-trust-store-pkcs11="pkcs11:"`: This switch tells gnutls to use the PKCS #11 trust store as the default trust. Omit this switch if [p11-kit-0.25.5](#) is not installed.

`--with-default-trust-store-file=/etc/pki/tls/certs/ca-bundle.crt`: This switch tells `configure` where to find the legacy CA certificate bundle and to use it instead of PKCS #11 module by default. Use this if [p11-kit-0.25.5](#) is not installed.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

`--enable-openssl-compatibility`: Use this switch if you wish to build the OpenSSL compatibility library.

`--without-p11-kit`: use this switch if you have not installed p11-kit.

`--with-included-unistring`: uses the bundled version of libunistring, instead of the system one. Use this switch if you have not installed [libunistring-1.2](#).

`--disable-dsa`: completely disables DSA algorithm support.

Contents

Installed Programs: certtool, danetool, gnutls-cli, gnutls-cli-debug, gnutls-serv, ocsptool, p11tool, psktool, and srptool

Installed Libraries: libgnutls.so, libgnutls-dane.so, libgnutlsxx.so, and libgnutls-openssl.so (optional)

Installed Directories: /usr/include/gnutls and /usr/share/doc/gnutls-3.8.7.1

Short Descriptions

<code>certtool</code>	is used to generate X.509 certificates, certificate requests, and private keys
<code>danetool</code>	is a tool used to generate and check DNS resource records for the DANE protocol
<code>gnutls-cli</code>	is a simple client program to set up a TLS connection to some other computer
<code>gnutls-cli-debug</code>	is a simple client program to set up a TLS connection to some other computer and produces very verbose progress results
<code>gnutls-serv</code>	is a simple server program that listens to incoming TLS connections
<code>ocsptool</code>	is a program that can parse and print information about OCSP requests/responses, generate requests and verify responses
<code>p11tool</code>	is a program that allows handling data from PKCS #11 smart cards and security modules
<code>psktool</code>	is a simple program that generates random keys for use with TLS-PSK

<code>srtplib</code>	is a simple program that emulates the programs in the Stanford SRP (Secure Remote Password) libraries using GnuTLS
<code>libgnutls.so</code>	contains the core API functions and X.509 certificate API functions

GPGME-1.23.2

Introduction to GPGME

The GPGME package is a C library that allows cryptography support to be added to a program. It is designed to make access to public key crypto engines like GnuPG or GpgSM easier for applications. GPGME provides a high-level crypto API for encryption, decryption, signing, signature verification and key management.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/gpgme/gpgme-1.23.2.tar.bz2>
- Download MD5 sum: 01a8c05b409847e87daf0543e91f8c37
- Download size: 1.8 MB
- Estimated disk space required: 260 MB (with tests)
- Estimated build time: 1.0 SBU (with all bindings and tests; with parallelism=4)

GPGME Dependencies

Required

[libassuan-3.0.1](#)

Optional

[Doxygen-1.12.0](#) and [Graphviz-12.1.0](#) (for API documentation), [GnuPG-2.4.5](#) (required if Qt or SWIG are installed; used during the test suite), [Clisp-2.49](#), [qt5-components-5.15.14](#), and [SWIG-4.2.1](#) (for language bindings)

Installation of GPGME

Install GPGME by running the following commands:

```
mkdir build &&
cd build &&

../configure --prefix=/usr --disable-gpg-test &&
make PYTHONS=
```

If [SWIG-4.2.1](#) is installed, build the Python 3 binding as a wheel:

```
if swig -version > /dev/null; then
    srccdir=$PWD/./lang/python \
    top_builddir=$PWD \
    pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD/lang/python
fi
```

To test the results, you should have [GnuPG-2.4.5](#) installed and remove the `--disable-gpg-test` above. If [SWIG-4.2.1](#) is installed, it's necessary to adapt the test suite to use the Python 3 binding just built as a wheel as well. Issue:

```
if swig -version > /dev/null; then
    python3 -m venv testenv
    testenv/bin/pip3 install --no-index --find-links=dist --no-cache-dir \
        gpg \
        sed '/PYTHON/s#run-tests.py#& --python-libdir=/dev/null#' \
        -i lang/python/tests/Makefile
fi &&

make -k check PYTHONS= PYTHON=$PWD/testenv/bin/python3
```

One test named `t-quick-key-manipulation.py` is known to fail.

Now, as the `root` user:

```
make install PYTHONS=
```

If [SWIG-4.2.1](#) is installed, still as the `root` user, install the Python 3 binding:

```
if swig -version > /dev/null; then
    pip3 install --no-index --find-links=dist --no-cache-dir --no-user gpg
fi
```

Command Explanations

`--disable-gpg-test`: if this parameter is not passed to configure, the test programs are built during `make` stage, which requires [GnuPG-2.4.5](#). This parameter is not needed if [GnuPG-2.4.5](#) is installed.

`PYTHONS`: Disable building Python binding using the deprecated `python3 setup.py build` command. The explicit instruction to build the Python 3 binding with the `pip3 wheel` command is provided.

Contents

Installed Program: gpgme-json, and gpgme-tool

Installed Libraries: libgpgme.so, libgpgmepp.so, and libqqpgme.so

Installed Directory: /usr/include/{gpgme++,qpgpgme,QGpgME}, /usr/lib/cmake/{Gpgmepp,QGpgme}.
/usr/lib/python3.12/site-packages/gpg{,-1.23.2.dist-info}, and /usr/share/common-lisp/source/gpgme

Short Descriptions

<code>gpgme-json</code>	outputs GPGME commands in JSON format
<code>gpgme-tool</code>	is an assuan server exposing GPGME operations, such as printing fingerprints and keyids with keyservers
<code>libgpgme.so</code>	contains the GPGME API functions
<code>libgpgmepp.so</code>	contains the C++ GPGME API functions
<code>libqqpgme.so</code>	contains API functions for handling GPG operations in Qt applications

iptables-1.8.10

Introduction to iptables

iptables is a userspace command line program used to configure the Linux 2.4 and later kernel packet filtering ruleset.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.netfilter.org/projects/iptables/files/iptables-1.8.10.tar.xz>
- Download MD5 sum: 5eaa3bb424dd3a13c98c0cb026314029
- Download size: 628 KB
- Estimated disk space required: 16 MB
- Estimated build time: 0.1 SBU

iptables Dependencies

Optional

[libpcap-1.10.4](#) (required for BPF compiler or nfproxy support), [bpf-utils](#) (required for Berkeley Packet Filter support), [libnfnetlink](#) (required for connlabel support), [libnetfilter_conntrack](#) (required for connlabel support), and [nftables](#)

Kernel Configuration

A firewall in Linux is accomplished through the netfilter interface. To use iptables to configure netfilter, the following kernel configuration parameters are required:

```
[*] Networking support ---> [NET]
Networking options --->
[*] Network packet filtering framework (Netfilter) ---> [NETFILTER]
```

```

[*] Advanced netfilter configuration           [NETFILTER_ADVANCED]
Core Netfilter Configuration --->
  <*/M> Netfilter connection tracking support      [NF_CONNTRACK]
  <*/M> Netfilter Xtables support (required for ip_tables)
    ...
  <*/M> LOG target support                   [NETFILTER_XT_TARGET_LOG]
IP: Netfilter Configuration --->
  <*/M> IP tables support (required for filtering/masq/NAT)
    ...
    ... [IP_NF_IPTABLES]

```

Include any connection tracking protocols that will be used, as well as any protocols that you wish to use for match support under the "Core Netfilter Configuration" section. The above options are enough for running [Creating a Personal Firewall With iptables](#) below.

Installation of iptables

Note

The installation below does not include building some specialized extension libraries which require the raw headers in the Linux source code. If you wish to build the additional extensions (if you aren't sure, then you probably don't), you can look at the `INSTALL` file to see an example of how to change the `KERNEL_DIR=` parameter to point at the Linux source code. Note that if you upgrade the kernel version, you may also need to recompile iptables and that the BLFS team has not tested using the raw kernel headers.

Install iptables by running the following commands:

```

./configure --prefix=/usr      \
--disable-nftables \
--enable-libipq   &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-nftables`: This switch disables building nftables compatibility.

`--enable-libipq`: This switch enables building of `libipq.so` which can be used by some packages outside of BLFS.

`--enable-nfsynproxy`: This switch enables installation of nfsynproxy SYNPROXY configuration tool.

Configuring iptables

Note

In the following example configurations, **LAN1** is used for the internal LAN interface, and **WAN1** is used for the external interface connected to the Internet. You will need to replace these values with appropriate interface names for your system.

Personal Firewall

A Personal Firewall is designed to let you access all the services offered on the Internet while keeping your computer secure and your data private.

Below is a slightly modified version of Rusty Russell's recommendation from the [Linux 2.4 Packet Filtering HOWTO](#). It is still applicable to the Linux 6.x kernels.

```

install -v -dm755 /etc/systemd/scripts
cat > /etc/systemd/scripts/iptables << "EOF"
#!/bin/sh

```

```

# Begin /etc/systemd/scripts/iptables

# Insert connection-tracking modules
# (not needed if built into the kernel)
modprobe nf_conntrack
modprobe xt_LOG

# Enable broadcast echo Protection
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

# Disable Source Routed Packets
echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route
echo 0 > /proc/sys/net/ipv4/conf/default/accept_source_route

# Enable TCP SYN Cookie Protection
echo 1 > /proc/sys/net/ipv4/tcp_syncookies

# Disable ICMP Redirect Acceptance
echo 0 > /proc/sys/net/ipv4/conf/default/accept_redirects

# Do not send Redirect Messages
echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
echo 0 > /proc/sys/net/ipv4/conf/default/send_redirects

# Drop Spoofed Packets coming in on an interface, where responses
# would result in the reply going out a different interface.
echo 1 > /proc/sys/net/ipv4/conf/all/rp_filter
echo 1 > /proc/sys/net/ipv4/conf/default/rp_filter

# Log packets with impossible addresses.
echo 1 > /proc/sys/net/ipv4/conf/all/log_martians
echo 1 > /proc/sys/net/ipv4/conf/default/log_martians

# be verbose on dynamic ip-addresses (not needed in case of static IP)
echo 2 > /proc/sys/net/ipv4/ip_dynaddr

# disable Explicit Congestion Notification
# too many routers are still ignorant
echo 0 > /proc/sys/net/ipv4/tcp_ecn

# Set a known state
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP

# These lines are here in case rules are already in place and the
# script is ever rerun on the fly. We want to remove all rules and
# pre-existing user defined chains before we implement new rules.
iptables -F
iptables -X
iptables -Z

iptables -t nat -F

# Allow local-only connections
iptables -A INPUT -i lo -j ACCEPT

# Free output on any interface to any ip for any service
# (equal to -P ACCEPT)
iptables -A OUTPUT -j ACCEPT

# Permit answers on already established connections
# and permit new connections related to established ones
# (e.g. port mode ftp)
iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

# Log everything else.
iptables -A INPUT -j LOG --log-prefix "FIREWALL:INPUT"

# End /etc/systemd/scripts/iptables
EOF
chmod 700 /etc/systemd/scripts/iptables

```

This script is quite simple, it drops all traffic coming into your computer that wasn't initiated from your computer, but as long as you are simply surfing the Internet you are unlikely to exceed its limits.

If you frequently encounter certain delays at accessing FTP servers, take a look at [BusyBox with iptables example number 4](#).

Even if you have daemons or services running on your system, these will be inaccessible everywhere but from your computer itself. If you want to allow access to services on your machine, such as `ssh` or `ping`, take a look at [Creating a BusyBox With iptables](#).

Masquerading Router

A Network Firewall has two interfaces, one connected to an intranet, in this example **LAN1**, and one connected to the Internet, here **WAN1**. To provide the maximum security for the firewall itself, make sure that there are no unnecessary servers running on it such as X11. As a general principle, the firewall itself should not access any untrusted service (think of a remote server giving answers that makes a daemon on your system crash, or even worse, that implements a worm via a buffer-overflow).

```
install -v -dm755 /etc/systemd/scripts

cat > /etc/systemd/scripts/iptables << "EOF"
#!/bin/sh

# Begin /etc/systemd/scripts/iptables

echo
echo "You're using the example configuration for a setup of a firewall"
echo "from Beyond Linux From Scratch."
echo "This example is far from being complete, it is only meant"
echo "to be a reference."
echo "Firewall security is a complex issue, that exceeds the scope"
echo "of the configuration rules below."

echo "You can find additional information"
echo "about firewalls in Chapter 4 of the BLFS book."
echo "https://www.linuxfromscratch.org/blfs"
echo

# Insert iptables modules (not needed if built into the kernel).

modprobe nf_conntrack
modprobe nf_conntrack_ftp
modprobe xt_conntrack
modprobe xt_LOG
modprobe xt_state

# Enable broadcast echo Protection
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

# Disable Source Routed Packets
echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route

# Enable TCP SYN Cookie Protection
echo 1 > /proc/sys/net/ipv4/tcp_syncookies

# Disable ICMP Redirect Acceptance
echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects

# Don't send Redirect Messages
echo 0 > /proc/sys/net/ipv4/conf/default/send_redirects

# Drop Spoofed Packets coming in on an interface where responses
# would result in the reply going out a different interface.
echo 1 > /proc/sys/net/ipv4/conf/default/rp_filter

# Log packets with impossible addresses.
echo 1 > /proc/sys/net/ipv4/conf/all/log_martians

# Be verbose on dynamic ip-addresses (not needed in case of static IP)
echo 2 > /proc/sys/net/ipv4/ip_dynaddr

# Disable Explicit Congestion Notification
# Too many routers are still ignorant
echo 0 > /proc/sys/net/ipv4/tcp_ecn

# Set a known state
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP
```

```

# These lines are here in case rules are already in place and the
# script is ever rerun on the fly. We want to remove all rules and
# pre-existing user defined chains before we implement new rules.
iptables -F
iptables -X
iptables -Z

iptables -t nat -F

# Allow local connections
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Allow forwarding if the initiated on the intranet
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD ! -i WAN1 -m conntrack --ctstate NEW -j ACCEPT

# Do masquerading
# (not needed if intranet is not using private ip-addresses)
iptables -t nat -A POSTROUTING -o WAN1 -j MASQUERADE

# Log everything for debugging
# (last of all rules, but before policy rules)
iptables -A INPUT -j LOG --log-prefix "FIREWALL:INPUT"
iptables -A FORWARD -j LOG --log-prefix "FIREWALL:FORWARD"
iptables -A OUTPUT -j LOG --log-prefix "FIREWALL:OUTPUT"

# Enable IP Forwarding
echo 1 > /proc/sys/net/ipv4/ip_forward

# The following sections allow inbound packets for specific examples
# Uncomment the example lines and adjust as necessary

# Allow ping on the external interface
#iptables -A INPUT -p icmp -m icmp --icmp-type echo-request -j ACCEPT
#iptables -A OUTPUT -p icmp -m icmp --icmp-type echo-reply -j ACCEPT

# Reject ident packets with TCP reset to avoid delays with FTP or IRC
#iptables -A INPUT -p tcp --dport 113 -j REJECT --reject-with tcp-reset

# Allow HTTP and HTTPS to 192.168.0.2
#iptables -A PREROUTING -t nat -i WAN1 -p tcp --dport 80 -j DNAT --to 192.168.0.2
#iptables -A PREROUTING -t nat -i WAN1 -p tcp --dport 443 -j DNAT --to 192.168.0.2
#iptables -A FORWARD -p tcp -d 192.168.0.2 --dport 80 -j ACCEPT
#iptables -A FORWARD -p tcp -d 192.168.0.2 --dport 443 -j ACCEPT

# End /etc/systemd/scripts/iptables
EOF
chmod 700 /etc/systemd/scripts/iptables

```

With this script your intranet should be reasonably secure against external attacks. No one should be able to setup a new connection to any internal service and, if it's masqueraded, makes your intranet invisible to the Internet. Furthermore, your firewall should be relatively safe because there are no services running that a cracker could attack.

BusyBox

This scenario isn't too different from the [Creating a Masquerading Router With iptables](#), but additionally offers some services to your intranet. Examples of this can be when you want to administer your firewall from another host on your intranet or use it as a proxy or a name server.

Note

Outlining specifically how to protect a server that offers services on the Internet goes far beyond the scope of this document. See the references in [the section called "Extra Information"](#) for more information.

Be cautious. Every service you have enabled makes your setup more complex and your firewall less secure. You are exposed to the risks of misconfigured services or running a service with an exploitable bug. A firewall should generally not run any extra services. See the introduction to the [Creating a Masquerading Router With iptables](#) for some more details.

If you want to add services such as internal Samba or name servers that do not need to access the Internet themselves, the additional statements are quite simple and should still be acceptable from a security standpoint. Just add the following lines into the script *before* the logging rules.

```
iptables -A INPUT -i ! WAN1 -j ACCEPT
iptables -A OUTPUT -o ! WAN1 -j ACCEPT
```

If daemons, such as squid, have to access the Internet themselves, you could open OUTPUT generally and restrict INPUT.

```
iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT -j ACCEPT
```

However, it is generally not advisable to leave OUTPUT unrestricted. You lose any control over trojans who would like to "call home", and a bit of redundancy in case you've (mis-)configured a service so that it broadcasts its existence to the world.

To accomplish this, you should restrict INPUT and OUTPUT on all ports except those that it's absolutely necessary to have open. Which ports you have to open depends on your needs: mostly you will find them by looking for failed accesses in your log files.

Have a Look at the Following Examples:

- Squid is caching the web:

```
iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --sport 80 -m conntrack --ctstate ESTABLISHED \
-j ACCEPT
```

- Your caching name server (e.g., named) does its lookups via UDP:

```
iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
```

- You want to be able to ping your computer to ensure it's still alive:

```
iptables -A INPUT -p icmp -m icmp --icmp-type echo-request -j ACCEPT
iptables -A OUTPUT -p icmp -m icmp --icmp-type echo-reply -j ACCEPT
```

- If you are frequently accessing FTP servers or enjoy chatting, you might notice delays because some implementations of these daemons query an identd daemon on your system to obtain usernames. Although there's really little harm in this, having an identd running is not recommended because many security experts feel the service gives out too much additional information.

To avoid these delays you could reject the requests with a 'tcp-reset' response:

```
iptables -A INPUT -p tcp --dport 113 -j REJECT --reject-with tcp-reset
```

- To log and drop invalid packets (packets that came in after netfilter's timeout or some types of network scans) insert these rules at the top of the chain:

```
iptables -I INPUT 0 -p tcp -m conntrack --ctstate INVALID \
-j LOG --log-prefix "FIREWALL:INVALID"
iptables -I INPUT 1 -p tcp -m conntrack --ctstate INVALID -j DROP
```

- Anything coming from the outside should not have a private address, this is a common attack called IP-spoofing:

```
iptables -A INPUT -i WAN1 -s 10.0.0.0/8 -j DROP
iptables -A INPUT -i WAN1 -s 172.16.0.0/12 -j DROP
iptables -A INPUT -i WAN1 -s 192.168.0.0/16 -j DROP
```

There are other addresses that you may also want to drop: 0.0.0.0/8, 127.0.0.0/8, 224.0.0.0/3 (multicast and experimental), 169.254.0.0/16 (Link Local Networks), and 192.0.2.0/24 (IANA defined test network).

- If your firewall is a DHCP client, you need to allow those packets:

```
iptables -A INPUT -i WAN1 -p udp -s 0.0.0.0 --sport 67 \
-d 255.255.255.255 --dport 68 -j ACCEPT
```

- To simplify debugging and be fair to anyone who'd like to access a service you have disabled, purposely or by mistake, you could REJECT those packets that are dropped.

Obviously this must be done directly after logging as the very last lines before the packets are dropped by policy:

```
iptables -A INPUT -j REJECT
```

These are only examples to show you some of the capabilities of the firewall code in Linux. Have a look at the man page of iptables. There you will find much more information. The port numbers needed for this can be found in /etc/services, in case you didn't find them by trial and error in your log file.

Systemd Unit

To set up the iptables firewall at boot, install the `iptables.service` unit included in the [blfs-systemd-units-20240801](#) package.

```
make install-iptables
```

Contents

- Installed Programs:** ip6tables, ip6tables-apply, ip6tables-legacy, ip6tables-legacy-restore, ip6tables-legacy-save, ip6tables-restore, ip6tables-save, iptables, iptables-apply, iptables-legacy, iptables-legacy-restore, iptables-legacy-apply, iptables-restore, iptables-save, iptables-xml, nfsynproxy (optional), and xtables-multi
- Installed Libraries:** libip4tc.so, libip6tc.so, libipq.so, libiptc.so, and libxtables.so
- Installed Directories:** /lib/xtables and /usr/include/libiptc

Short Descriptions

<code>iptables</code>	is used to set up, maintain, and inspect the tables of IP packet filter rules in the Linux kernel
<code>iptables-apply</code>	is a safer way to update iptables remotely
<code>iptables-legacy</code>	is used to interact with iptables using the legacy command set
<code>iptables-legacy-restore</code>	is used to restore a set of legacy iptables rules
<code>iptables-legacy-save</code>	is used to save a set of legacy iptables rules
<code>iptables-restore</code>	is used to restore IP Tables from data specified on STDIN. Use I/O redirection provided by your shell to read from a file
<code>iptables-save</code>	is used to dump the contents of an IP Table in easily parseable format to STDOUT. Use I/O redirection provided by your shell to write to a file
<code>iptables-xml</code>	is used to convert the output of <code>iptables-save</code> to an XML format. Using the <code>iptables.xslt</code> stylesheet converts the XML back to the format of <code>iptables-restore</code>
<code>ip6tables*</code>	are a set of commands for IPV6 that parallel the iptables commands above
<code>nfsynproxy</code>	(optional) configuration tool. SYNPROXY target makes handling of large SYN floods possible without the large performance penalties imposed by the connection tracking in such cases
<code>xtables-multi</code>	is a binary that behaves according to the name it is called by

Setting Up a Network Firewall

Introduction to Firewall Creation

The purpose of a firewall is to protect a computer or a network against malicious access. In a perfect world every daemon or service, on every machine, is perfectly configured and immune to security flaws, and all users are trusted implicitly to use the equipment as intended. However, this is rarely, if ever, the case. Daemons may be misconfigured, or updates may not have been applied for known exploits against essential services. Additionally, you may wish to choose which services are accessible by certain machines or users, or you may wish to limit which machines or applications are allowed external access. Alternatively, you simply may not trust some of your applications or users. For these reasons, a carefully designed firewall should be an essential part of system security.

While a firewall can greatly limit the scope of the above issues, do not assume that having a firewall makes careful configuration redundant, or that any negligent misconfiguration is harmless. A firewall does not prevent the exploitation of any service you offer outside of it. Despite having a firewall, you need to keep applications and daemons properly configured and up to date.

Meaning of the Word "Firewall"

The word firewall can have several different meanings.

Personal Firewall

This is a hardware device or software program, intended to secure a home or desktop computer connected to the Internet. This type of firewall is highly relevant for users who do not know how their computers might be accessed via the Internet or how to disable that access, especially if they are always online and connected via broadband links.

An example configuration for a personal firewall is provided at [Creating a Personal Firewall With iptables](#).

Masquerading Router

This is a system placed between the Internet and an intranet. To minimize the risk of compromising the firewall itself, it should generally have only one role—that of protecting the intranet. Although not completely risk-free, the tasks of doing the routing and IP masquerading (rewriting IP headers of the packets it routes from clients with private IP addresses onto the Internet so that they seem to come from the firewall itself) are commonly considered relatively secure.

An example configuration for a masquerading firewall is provided at [Creating a Masquerading Router With iptables](#).

BusyBox

This is often an old computer you may have retired and nearly forgotten, performing masquerading or routing functions, but offering non-firewall services such as a web-cache or mail. This may be used for home networks, but is not to be considered as secure as a firewall only machine because the combination of server and router/firewall on one machine raises the complexity of the setup.

An example configuration for a BusyBox is provided at [Creating a BusyBox With iptables](#).

Firewall with a Demilitarized Zone

This type of firewall performs masquerading or routing, but grants public access to some branch of your network that is physically separated from your regular intranet and is essentially a separate network with direct Internet access. The servers on this network are those which must be easily accessible from both the Internet and intranet. The firewall protects both networks. This type of firewall has a minimum of three network interfaces.

Packetfilter

This type of firewall does routing or masquerading but does not maintain a state table of ongoing communication streams. It is fast but quite limited in its ability to block undesired packets without blocking desired packets.

Conclusion

Caution

The example configurations provided for [iptables-1.8.10](#) are not intended to be a complete guide to securing systems. Firewalling is a complex issue that requires careful configuration. The configurations provided by BLFS are intended only to give examples of how a firewall works. They are not intended to fit any particular configuration and may not provide complete protection from an attack.

BLFS provides an utility to manage the kernel Netfilter interface, [iptables-1.8.10](#). It has been around since early 2.4 kernels, and has been the standard since. This is likely the set of tools that will be most familiar to existing admins. Other tools have been developed more recently, see the list of further readings below for more details. Here you will find a list of URLs that contain comprehensive information about building firewalls and further securing your system.

Extra Information

Further Reading on Firewalls

www.netfilter.org - Homepage of the netfilter/iptables/nftables projects
[Netfilter related FAQ](#)
[Netfilter related HOWTO's](#)
[nftables HOWTO](#)
tldp.org/LDP/nag2/x-087-2-firewall.html
tldp.org/HOWTO/Security-HOWTO.html
tldp.org/HOWTO/Firewall-HOWTO.html
linuxsecurity.com/howtos
www.circlemud.org/jelson/writings/security/index.htm
insecure.org/reading.html

libcap-2.70 with PAM

Introduction to libcap with PAM

The libcap package was installed in LFS, but if Linux-PAM support is desired, the PAM module must be built (after installation of Linux-PAM).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/libs/security/linux-privs/libcap2/libcap-2.70.tar.xz>
- Download MD5 sum: df0e20c6eeaca849347b87d5d6a8870c0
- Download size: 188 KB
- Estimated disk space required: 2.2 MB
- Estimated build time: less than 0.1 SBU

libcap Dependencies

Required

[Linux-PAM-1.6.1](#)

Installation of libcap

Note

If you are upgrading libcap from a previous version, use the instructions in [LFS libcap page](#) to upgrade libcap. If [Linux-PAM-1.6.1](#) has been built, the PAM module will automatically be built too.

Install libcap by running the following commands:

```
make -C pam_cap
```

This package does not come with a test suite.

Now, as the `root` user:

```
install -v -m755 pam_cap/pam_cap.so /usr/lib/security &&
install -v -m644 pam_cap/capability.conf /etc/security
```

Configuring Libcap

In order to allow Linux-PAM to grant privileges based on POSIX capabilities, you need to add the libcap module to the beginning of the `/etc/pam.d/system-auth` file. Make the required edits with the following commands:

```
mv -v /etc/pam.d/system-auth{,.bak} &&
cat > /etc/pam.d/system-auth << "EOF" &&
# Begin /etc/pam.d/system-auth

auth      optional      pam_cap.so
EOF
tail -n +3 /etc/pam.d/system-auth.bak >> /etc/pam.d/system-auth
```

Additionally, you'll need to modify the `/etc/security/capability.conf` file to grant necessary privileges to users, and utilize the `setcap` utility to set capabilities on specific utilities as needed. See `man 8 setcap` and `man 3 cap_from_text` for additional information.

Contents

Installed Programs: None

Installed Library: pam_cap.so

Installed Directories: None

Linux-PAM-1.6.1

Introduction to Linux PAM

The Linux PAM package contains Pluggable Authentication Modules used by the local system administrator to control how application programs authenticate users.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/linux-pam/linux-pam/releases/download/v1.6.1/Linux-PAM-1.6.1.tar.xz>
- Download MD5 sum: 8ad1e72d1ff6480d8e0af658e2d7b768
- Download size: 1.0 MB
- Estimated disk space required: 39 MB (with tests)

- Estimated build time: 0.4 SBU (with tests)

Additional Downloads

Optional Documentation

- Download (HTTP): <https://github.com/linux-pam/linux-pam/releases/download/v1.6.1/Linux-PAM-1.6.1-docs.tar.xz>
- Download MD5 sum: 46dc9f9a27ef73a2fbe3b667877e88da
- Download size: 455 KB

Linux PAM Dependencies

Optional

[libnsl-2.0.1](#), [libtirpc-1.3.5](#), [rpcsvc-proto-1.4.4](#), [Berkeley DB](#) (deprecated), [libaudit](#), [libeconf](#), and [Prelude](#)

Note

[Shadow-4.16.0](#) and [Systemd-256.4](#) must be reinstalled and reconfigured after installing and configuring Linux PAM.

With Linux-PAM-1.4.0 and higher, the pam_cracklib module is not installed by default. Use [libpwquality-1.4.5](#) to enforce strong passwords.

Kernel Configuration

For the PAM module `pam_loginuid.so` (referred by the PAM configuration file `system-session` if [Systemd-256.4](#) is rebuilt with [PAM support](#) later) to work, a kernel configuration parameter need to be set or the module will just do nothing:

```
General setup --->
  [*] Auditing support                                [AUDIT]
```

Installation of Linux PAM

The shipped `libtool.m4` file has a configuration inconsistent with LFS `/usr` hierarchy. This issue would cause `libpam_misc.so` linked with an rpath flag which may sometimes cause troubles or even security issues. Regenerate the building system to fix the inconsistency:

```
autoreconf -fi
```

If you downloaded the documentation, unpack the tarball by issuing the following command.

```
tar -xf ../Linux-PAM-1.6.1-docs.tar.xz --strip-components=1
```

Compile and link Linux PAM by running the following commands:

```
./configure --prefix=/usr          \
            --sbin=/usr/sbin      \
            --sysconfdir=/etc       \
            --libdir=/usr/lib        \
            --enable-securedir=/usr/lib/security \
            --docdir=/usr/share/doc/Linux-PAM-1.6.1 &&
make
```

To test the results, a suitable `/etc/pam.d/other` configuration file must exist.

Reinstallation or Upgrade of Linux PAM

If you have a system with Linux PAM installed and working, be careful when modifying the files in `/etc/pam.d`, since your system may become totally unusable. If you want to run the tests, you do not need to create another `/etc/pam.d/other` file. The existing file can be used for the tests.

You should also be aware that `make install` overwrites the configuration files in `/etc/security` as well as `/etc/environment`. If you have modified those files, be sure to back them up.

For a first-time installation, create a configuration file by issuing the following commands as the `root` user:

```
install -v -m755 -d /etc/pam.d &&

cat > /etc/pam.d/other << "EOF"
auth      required      pam_deny.so
account   required      pam_deny.so
password  required      pam_deny.so
session   required      pam_deny.so
EOF
```

Now run the tests by issuing `make check`. Be sure the tests produced no errors before continuing the installation. Note that the tests are very long. Redirect the output to a log file, so you can inspect it thoroughly.

For a first-time installation, remove the configuration file created earlier by issuing the following command as the `root` user:

```
rm -fv /etc/pam.d/other
```

Now, as the `root` user:

```
make install &&
chmod -v 4755 /usr/sbin/unix_chkpwd
```

Command Explanations

`--enable-securedir=/usr/lib/security`: This switch sets the installation location for the PAM modules.

`chmod -v 4755 /usr/sbin/unix_chkpwd`: The setuid bit for the `unix_chkpwd` helper program must be turned on, so that non-`root` processes can access the shadow file.

Configuring Linux-PAM

Configuration Files

`/etc/security/*` and `/etc/pam.d/*`

Configuration Information

Configuration information is placed in `/etc/pam.d/`. Here is a sample file:

```
# Begin /etc/pam.d/other

auth      required      pam_unix.so      nullok
account   required      pam_unix.so
session   required      pam_unix.so
password  required      pam_unix.so      nullok

# End /etc/pam.d/other
```

Now create some generic configuration files. As the `root` user:

```
install -vdm755 /etc/pam.d &&
cat > /etc/pam.d/system-account << "EOF" &&
# Begin /etc/pam.d/system-account

account   required      pam_unix.so

# End /etc/pam.d/system-account
EOF

cat > /etc/pam.d/system-auth << "EOF" &&
# Begin /etc/pam.d/system-auth

auth      required      pam_unix.so

# End /etc/pam.d/system-auth
EOF
```

```

cat > /etc/pam.d/system-session << "EOF" &&
# Begin /etc/pam.d/system-session

session    required      pam_unix.so

# End /etc/pam.d/system-session
EOF

cat > /etc/pam.d/system-password << "EOF"
# Begin /etc/pam.d/system-password

# use yescrypt hash for encryption, use shadow, and try to use any
# previously defined authentication token (chosen password) set by any
# prior module.
password   required      pam_unix.so          yescrypt shadow try_first_pass

# End /etc/pam.d/system-password
EOF

```

If you wish to enable strong password support, install [libpwquality-1.4.5](#), and follow the instructions on that page to configure the pam_pwquality PAM module with strong password support.

Next, add a restrictive `/etc/pam.d/other` configuration file. With this file, programs that are PAM aware will not run unless a configuration file specifically for that application exists.

```

cat > /etc/pam.d/other << "EOF"
# Begin /etc/pam.d/other

auth      required      pam_warn.so
auth      required      pam_deny.so
account   required      pam_warn.so
account   required      pam_deny.so
password  required      pam_warn.so
password  required      pam_deny.so
session   required      pam_warn.so
session   required      pam_deny.so

# End /etc/pam.d/other
EOF

```

The PAM man page (`man pam`) provides a good starting point to learn about the several fields, and allowable entries. The [Linux-PAM System Administrators' Guide](#) is recommended for additional information.

Important

You should now reinstall the [Shadow-4.16.0](#) and [Systemd-256.4](#) packages.

Contents

Installed Program: faillock, mkhomedir_helper, pam_namespace_helper, pam_timestamp_check, pwhistory_helper, unix_chkpwd and unix_update

Installed Libraries: libpam.so, libpamc.so and libpam_misc.so

Installed Directories: /etc/security, /usr/lib/security, /usr/include/security and /usr/share/doc/Linux-PAM-1.6.1

Short Descriptions

<code>faillock</code>	displays and modifies the authentication failure record files
<code>mkhomedir_helper</code>	is a helper binary that creates home directories
<code>pam_namespace_helper</code>	is a helper program used to configure a private namespace for a user session
<code>pwhistory_helper</code>	is a helper program that transfers password hashes from passwd or shadow to opasswd
<code>pam_timestamp_check</code>	is used to check if the default timestamp is valid
<code>unix_chkpwd</code>	is a helper binary that verifies the password of the current user
<code>unix_update</code>	is a helper binary that updates the password of a given user
<code>libpam.so</code>	provides the interfaces between applications and the PAM modules

liboauth-1.0.3

Introduction to liboauth

liboauth is a collection of POSIX-C functions implementing the OAuth Core RFC 5849 standard. Liboauth provides functions to escape and encode parameters according to OAuth specification and offers high-level functionality to sign requests or verify OAuth signatures as well as perform HTTP requests.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/liboauth/liboauth-1.0.3.tar.gz>
- Download MD5 sum: 689b46c2b3ab1a39735ac33f714c4f7f
- Download size: 496 KB
- Estimated disk space required: 3.5 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch for use with openssl: <https://www.linuxfromscratch.org/patches/blfs/12.2/liboauth-1.0.3-openssl-1.1.0-3.patch>

liboauth Dependencies

Required

[curl-8.9.1](#)

Optional

[nss-3.103](#) and [doxygen-1.12.0](#) (to build documentation)

Installation of liboauth

Apply a patch for the current version of openssl:

```
patch -Np1 -i ../../liboauth-1.0.3-openssl-1.1.0-3.patch
```

Install liboauth by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

If you wish to build the documentation (needs [doxygen-1.12.0](#)), issue:

```
make dox
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you have previously built the documentation, install it by running the following commands as the `root` user:

```
install -v -dm755 /usr/share/doc/liboauth-1.0.3 &&  
cp -rv doc/html/* /usr/share/doc/liboauth-1.0.3
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-nss`: Use this switch if you want to use Mozilla NSS instead of OpenSSL.

Contents

Installed Programs: None

Installed Libraries: liboauth.so

Installed Directories: /usr/share/doc/liboauth-1.0.3

Short Descriptions

liboauth.so provides functions to escape and encode strings according to OAuth specifications and offers high-level functionality built on top to sign requests or verify signatures using either NSS or OpenSSL for calculating the hash/signatures

libpwquality-1.4.5

Introduction to libpwquality

The libpwquality package provides common functions for password quality checking and also scoring them based on their apparent randomness. The library also provides a function for generating random passwords with good pronounceability.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libpwquality/libpwquality/releases/download/libpwquality-1.4.5/libpwquality-1.4.5.tar.bz2>
- Download MD5 sum: 6b70e355269aef0b9ddb2b9d17936f21
- Download size: 424 KB
- Estimated disk space required: 5.4 MB
- Estimated build time: 0.1 SBU

libpwquality Dependencies

Required

[CrackLib-2.10.2](#)

Recommended

[Linux-PAM-1.6.1](#)

Installation of libpwquality

Install libpwquality by running the following commands:

```
./configure --prefix=/usr           \
            --disable-static        \
            --with-securedir=/usr/lib/security \
            --disable-python-bindings    &&
make &&
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD/python
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pwquality
```

Command Explanations

`--disable-python-bindings`: This parameter disables building Python bindings with the deprecated `python3 setup.py build` command. The explicit instruction to build the Python 3 binding with the `pip3 wheel` command is provided.

Configuring libpwquality

`libpwquality` is intended to be a functional replacement for the now-obsolete `pam_cracklib.so` PAM module. To configure the system to use the `pam_pwquality` module, execute the following commands as the `root` user:

```
mv /etc/pam.d/system-password{,.orig} &&
cat > /etc/pam.d/system-password << "EOF"
# Begin /etc/pam.d/system-password

# check new passwords for strength (man pam_pwquality)
password required pam_pwquality.so authtok_type=UNIX retry=1 difok=1 \
    minlen=8 dcredit=0 ucredit=0 \
    lcredit=0 ocredit=0 minclass=1 \
    maxrepeat=0 maxsequence=0 \
    maxclassrepeat=0 gecoscheck=0 \
    dictcheck=1 usercheck=1 \
    enforcing=1 badwords="" \
    dictpath=/usr/lib/cracklib/pw_dict

# use yescrypt hash for encryption, use shadow, and try to use any
# previously defined authentication token (chosen password) set by any
# prior module.
password required pam_unix.so      yescrypt shadow try_first_pass

# End /etc/pam.d/system-password
EOF
```

Contents

Installed Programs: `pwscore` and `pwmake`

Installed Libraries: `pam_pwquality.so` and `libpwquality.so`

Installed Directories: `/usr/lib/python3.11/site-packages/pwquality-1.4.5.dist-info`

Short Descriptions

<code>pwmake</code>	is a simple configurable tool for generating random and relatively easily pronounceable passwords
<code>pwscore</code>	is a simple tool for checking quality of a password
<code>libpwquality.so</code>	contains API functions for checking the password quality
<code>pam_pwquality.so</code>	is a Linux PAM module used to perform password quality checking

MIT Kerberos V5-1.21.3

Introduction to MIT Kerberos V5

MIT Kerberos V5 is a free implementation of Kerberos 5. Kerberos is a network authentication protocol. It centralizes the authentication database and uses kerberized applications to work with servers or services that support Kerberos allowing single logins and encrypted communication over internal networks or the Internet.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://kerberos.org/dist/krb5/1.21/krb5-1.21.3.tar.gz>
- Download MD5 sum: beb34d1dfc72ba0571ce72bed03e06eb
- Download size: 8.7 MB
- Estimated disk space required: 95 MB (add 14 MB for tests)
- Estimated build time: 0.3 SBU (Using parallelism=4; add 1.1 SBU for tests)

MIT Kerberos V5 Dependencies

Optional

`BIND Utilities-9.20.0`, `CrackLib-2.10.2` (`/usr/share/dict/words` referred by some tests), `GnuPG-2.4.5` (to authenticate the package), `keyutils-1.6.3`, `OpenLDAP-2.6.8`, `Valgrind-3.23.0` (used during the test suite), `yasm-1.3.0`, `libedit`, `cmocka`, `kdcproxy`, `pyrad`, and `resolv_wrapper`

Note

Some sort of time synchronization facility on your system (like [ntp-4.2.8p18](#)) is required since Kerberos won't authenticate if there is a time difference between a kerberized client and the KDC server.

Installation of MIT Kerberos V5

Build MIT Kerberos V5 by running the following commands:

```
cd src &&
sed -i -e '/eq 0/{N;s/12 //}' plugins/kdb/db2/libdb2/test/run.test &&

./configure --prefix=/usr \
            --sysconfdir=/etc \
            --localstatedir=/var/lib \
            --runstatedir=/run \
            --with-system-et \
            --with-system-ss \
            --with-system-verto=no \
            --enable-dns-for-realm \
            --disable-rpath &&
make
```

To test the build, issue: `make -j1 -k check`. Some tests may fail with the latest version of dejagnu and glibc. Some tests may hang for a long time and fail if the system is not connected to a network. One test, `t_kadm5srv`, is known to fail. If [keyutils-1.6.3](#) is installed but [Keyutils Kernel Configuration](#) is not satisfied, some tests will fail complaining `keyctl failed with code 1`.

Now, as the `root` user:

```
make install &&
cp -vfr ./doc -T /usr/share/doc/krb5-1.21.3
```

Command Explanations

The `sed` command removes a test that is known to fail.

`--localstatedir=/var/lib`: This option is used so that the Kerberos variable runtime data is located in `/var/lib` instead of `/usr/var`.

`--runstatedir=/run`: This option is used so that the Kerberos runtime state information is located in `/run` instead of the deprecated `/var/run`.

`--with-system-et`: This switch causes the build to use the system-installed versions of the error-table support software.

`--with-system-ss`: This switch causes the build to use the system-installed versions of the subsystem command-line interface software.

`--with-system-verto=no`: This switch fixes a bug in the package: it does not recognize its own verto library installed previously. This is not a problem, if reinstalling the same version, but if you are updating, the old library is used as system's one, instead of installing the new version.

`--enable-dns-for-realm`: This switch allows realms to be resolved using the DNS server.

`--disable-rpath`: This switch prevents hard coding library search paths (rpath) into the binary executable files and shared libraries. This package does not need rpath for an installation into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`--with-ldap`: Use this switch if you want to compile the OpenLDAP database backend module.

Configuring MIT Kerberos V5

Config Files

`/etc/krb5.conf` and `/var/lib/krb5kdc/kdc.conf`

Configuration Information

Kerberos Configuration

Tip

You should consider installing some sort of password checking dictionary so that you can configure the installation to only accept strong passwords. A suitable dictionary to use is shown in the [CrackLib-2.10.2](#) instructions. Note that only one file can be used, but you can concatenate many files into one. The configuration file shown below assumes you have installed a dictionary to `/usr/share/dict/words`.

Create the Kerberos configuration file with the following commands issued by the `root` user:

```
cat > /etc/krb5.conf << "EOF"
# Begin /etc/krb5.conf

[libdefaults]
    default_realm = <EXAMPLE.ORG>
    encrypt = true

[realms]
    <EXAMPLE.ORG> = {
        kdc = <belgarath.example.org>
        admin_server = <belgarath.example.org>
        dict_file = /usr/share/dict/words
    }

[domain_realm]
    .<example.org> = <EXAMPLE.ORG>

[logging]
    kdc = SYSLOG:INFO:AUTH
    admin_server = SYSLOG:INFO:AUTH
    default = SYSLOG:DEBUG:DAEMON

# End /etc/krb5.conf
EOF
```

You will need to substitute your domain and proper hostname for the occurrences of the `<belgarath>` and `<example.org>` names.

`default_realm` should be the name of your domain changed to ALL CAPS. This isn't required, but both Heimdal and MIT recommend it.

`encrypt = true` provides encryption of all traffic between kerberized clients and servers. It's not necessary and can be left off. If you leave it off, you can encrypt all traffic from the client to the server using a switch on the client program instead.

The `[realms]` parameters tell the client programs where to look for the KDC authentication services.

The `[domain_realm]` section maps a domain to a realm.

Create the KDC database:

```
kdb5_util create -r <EXAMPLE.ORG> -s
```

Now you should populate the database with principals (users). For now, just use your regular login name or `root`.

```
kadmin.local
kadmin.local: add_policy dict-only
kadmin.local: addprinc -policy dict-only <loginname>
```

The KDC server and any machine running kerberized server daemons must have a host key installed:

```
kadmin.local: addprinc -randkey host/<belgarath.example.org>
```

After choosing the defaults when prompted, you will have to export the data to a keytab file:

```
kadmin.local: ktadd host/<belgarath.example.org>
```

This should have created a file in `/etc` named `krb5.keytab` (Kerberos 5). This file should have 600 (`root rw only`) permissions. Keeping the keytab files from public access is crucial to the overall security of the Kerberos installation.

Exit the `kadmin` program (use `quit` or `exit`) and return back to the shell prompt. Start the KDC daemon manually, just to test out the installation:

```
/usr/sbin/krb5kdc
```

Attempt to get a ticket with the following command:

```
kinit <loginname>
```

You will be prompted for the password you created. After you get your ticket, you can list it with the following command:

```
klist
```

Information about the ticket should be displayed on the screen.

To test the functionality of the keytab file, issue the following command as the `root` user:

```
ktutil
ktutil: rkt /etc/krb5.keytab
ktutil: l
```

This should dump a list of the host principal, along with the encryption methods used to access the principal.

Create an empty ACL file that can be modified later:

```
touch /var/lib/krb5kdc/kadm5.acl
```

At this point, if everything has been successful so far, you can feel fairly confident in the installation and configuration of the package.

Additional Information

For additional information consult the [documentation for krb5-1.21.3](#) on which the above instructions are based.

Systemd Unit

If you want to start Kerberos services at boot, install the `krb5.service` unit included in the [blfs-systemd-units-20240801](#) package using the following command:

```
make install-krb5
```

Contents

- Installed Programs:** gss-client, gss-server, k5srvutil, kadmin, kadmin.local, kadminmind, kdb5_ldap_util (optional), kdb5_util, kdestroy, kinit, klist, kpasswd, kprop, kpropd, kproplog, krb5-config, krb5-send-pr, krb5kdc, ksu, kswitch, ktutil, kvno, scclient, sim_client, sim_server, sserver, uuclient, and uuserver
- Installed Libraries:** libgssapi_krb5.so, libgssrpc.so, libk5crypto.so, libkadm5clnt_mit.so, libkadm5clnt.so, libkadm5srv_mit.so, libkadm5srv.so, libkdb_ldap.so (optional), libkdb5.so, libkrad.so, libkrb5.so, libkrb5support.so, libverto.so, and some plugins under the /usr/lib/krb5 tree
- Installed Directories:** /usr/include/{gssapi,gssrpc,kadm5,krb5}, /usr/lib/krb5, /usr/share/{doc/krb5-1.21.3,examples/krb5}, /var/lib/krb5kdc, and /run/krb5kdc

Short Descriptions

<code>gss-client</code>	is a GSSAPI test client
<code>gss-server</code>	is a GSSAPI test server
<code>k5srvutil</code>	is a host keytable manipulation utility
<code>kadmin</code>	is an utility used to make modifications to the Kerberos database
<code>kadmin.local</code>	is an utility similar to <code>kadmin</code> , but if the database is db2, the local client <code>kadmin.local</code> , is intended to run directly on the master KDC without Kerberos authentication
<code>kadminmind</code>	is a server for administrative access to a Kerberos database
<code>kdb5_ldap_util</code> (optional)	allows an administrator to manage realms, Kerberos services and ticket policies
<code>kdb5_util</code>	is the KDC database utility
<code>kdestroy</code>	removes the current set of tickets
<code>kinit</code>	is used to authenticate to the Kerberos server as a principal and acquire a ticket granting ticket that can later be used to obtain tickets for other services
<code>klist</code>	reads and displays the current tickets in the credential cache
<code>kpasswd</code>	is a program for changing Kerberos 5 passwords

kprop	takes a principal database in a specified format and converts it into a stream of database records
kpropd	receives a database sent by kprop and writes it as a local database
kproplog	displays the contents of the KDC database update log to standard output
krb5-config	gives information on how to link programs against libraries
krb5kdc	is the Kerberos 5 server
krb5-send-pr	sends a problem report (PR) to a central support site
ksu	is the super user program using Kerberos protocol. Requires a properly configured <code>/etc/shells</code> and <code>~/.k5login</code> containing principals authorized to become super users
kswitch	makes the specified credential cache the primary cache for the collection, if a cache collection is available
ktutil	is a program for managing Kerberos keytabs
kvno	prints keyversion numbers of Kerberos principals
sclient	is used to contact a sample server and authenticate to it using Kerberos 5 tickets, then display the server's response
sim_client	is a simple UDP-based sample client program, for demonstration
sim_server	is a simple UDP-based server application, for demonstration
sserver	is the sample Kerberos 5 server
uuclient	is another sample client
uuserver	is another sample server
libgssapi_krb5.so	contains the Generic Security Service Application Programming Interface (GSSAPI) functions which provides security services to callers in a generic fashion, supportable with a range of underlying mechanisms and technologies and hence allowing source-level portability of applications to different environments
libkadm5clnt.so	contains the administrative authentication and password checking functions required by Kerberos 5 client-side programs
libkadm5srv.so	contains the administrative authentication and password checking functions required by Kerberos 5 servers
libkdb5.so	is a Kerberos 5 authentication/authorization database access library
libkrad.so	contains the internal support library for RADIUS functionality
libkrb5.so	is an all-purpose Kerberos 5 library

Nettle-3.10

Introduction to Nettle

The Nettle package contains a low-level cryptographic library that is designed to fit easily in many contexts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/nettle/nettle-3.10.tar.gz>
- Download MD5 sum: c61453139d5fb44e9cdcc5b684b26e55
- Download size: 2.5 MB
- Estimated disk space required: 100 MB (with tests)
- Estimated build time: 0.3 SBU (with tests; both using parallelism=4)

Nettle Dependencies

Optional

[Valgrind-3.23.0](#) (optional for the tests)

Installation of Nettle

Install Nettle by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
chmod -v 755 /usr/lib/lib{hogweed,nettle}.so &&
install -v -m755 -d /usr/share/doc/nettle-3.10 &&
install -v -m644 nettle.{html, pdf} /usr/share/doc/nettle-3.10
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: nettle-hash, nettle-lfib-stream, nettle-pbkdf2, pkcs1-conv and SEXP-conv

Installed Libraries: libhogweed.so and libnettle.so

Installed Directory: /usr/include/nettle and /usr/share/doc/nettle-3.10

Short Descriptions

<code>nettle-hash</code>	calculates a hash value using a specified algorithm
<code>nettle-lfib-stream</code>	outputs a sequence of pseudorandom (non-cryptographic) bytes, using Knuth's lagged fibonacci generator. The stream is useful for testing, but should not be used to generate cryptographic keys or anything else that needs real randomness
<code>nettle-pbkdf2</code>	is a password-based key derivation function that takes a password or a passphrase as input and returns a strengthened password, which is protected against pre-computation attacks by using salting and other expensive computations.
<code>pkcs1-conv</code>	converts private and public RSA keys from PKCS #1 format to SEXP format
<code>sexp-conv</code>	converts an s-expression to a different encoding

NSS-3.103

Introduction to NSS

The Network Security Services (NSS) package is a set of libraries designed to support cross-platform development of security-enabled client and server applications. Applications built with NSS can support SSL v2 and v3, TLS, PKCS #5, PKCS #7, PKCS #11, PKCS #12, S/MIME, X.509 v3 certificates, and other security standards. This is useful for implementing SSL and S/MIME or other Internet security standards into an application.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://archive.mozilla.org/pub/security/nss/releases/NSS_3_103_RTM/src/nss-3.103.tar.gz
- Download MD5 sum: 2823082a44b9dd71d6281108e0bab03f
- Download size: 73 MB
- Estimated disk space required: 304 MB (add 149 MB for tests)
- Estimated build time: 0.8 SBU (with parallelism=4, add 16 SBU for tests on AMD Ryzens or at least 30 SBU on Intel machines)

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/nss-3.103-standalone-1.patch>

NSS Dependencies

Required

[NSPR-4.35](#)

Recommended

[SQLite-3.46.1](#) and [p11-kit-0.25.5](#) (runtime)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/nss>

Installation of NSS

Install NSS by running the following commands:

```
patch -Np1 -i ../nss-3.103-standalone-1.patch &&  
  
cd nss &&  
  
make BUILD_OPT=1  
NSPR_INCLUDE_DIR=/usr/include/nspr \  
USE_SYSTEM_ZLIB=1  
ZLIB_LIBS=-lz  
NSS_ENABLE_WERROR=0  
$([ $(uname -m) = x86_64 ] && echo USE_64=1) \  
$([ -f /usr/include/sqlite3.h ] && echo NSS_USE_SYSTEM_SQLITE=1)
```

To run the tests, execute the following commands:

```
cd tests &&  
HOST=localhost DOMSUF=localdomain ./all.sh  
cd ../
```

Note

Some information about the tests:

- HOST=localhost and DOMSUF=localdomain are required. Without these variables, a FQDN is required to be specified and this generic way should work for everyone, provided `localhost.localdomain` is defined by the `myhostname` Name Service Switch module, as specified in [the LFS book](#).
- The tests take a long time to run. If desired there is information in the all.sh script about running subsets of the total test suite.
- When interrupting the tests, the test suite fails to spin down test servers that are run. This leads to an infinite loop in the tests where the test suite tries to kill a server that doesn't exist anymore because it pulls the wrong PID.
- Test suite results (in HTML format!) can be found at `..../test_results/security/localhost.1/results.html`
- A few tests might fail on some Intel machines for unknown reasons.

Now, as the `root` user:

```
cd ../dist &&  
  
install -v -m755 Linux*/lib/*.so          /usr/lib &&  
install -v -m644 Linux*/lib/{*.chk,libcrmf.a} /usr/lib &&  
  
install -v -m755 -d                      /usr/include/nss &&  
cp -v -RL {public,private}/nss/*        /usr/include/nss &&  
  
install -v -m755 Linux*/bin/{certutil,nss-config,pk12util} /usr/bin &&  
  
install -v -m644 Linux*/lib/pkgconfig/nss.pc /usr/lib/pkgconfig
```

Command Explanations

`BUILD_OPT=1`: This option is passed to `make` so that the build is performed with no debugging symbols built into the binaries and the default compiler optimizations are used.

`NSPR_INCLUDE_DIR=/usr/include/nspr`: This option sets the location of the nspr headers.

`USE_SYSTEM_ZLIB=1`: This option is passed to `make` to ensure that the `libssl13.so` library is linked to the system installed zlib instead of the in-tree version.

`ZLIB_LIBS=-lz`: This option provides the linker flags needed to link to the system zlib.

`$([[$uname -m] = x86_64] && echo USE_64=1):` The `USE_64=1` option is required on `x86_64`, otherwise `make` will try (and fail) to create 32-bit objects. The `[$uname -m] = x86_64` test ensures it has no effect on a 32 bit system.

`([[-f /usr/include/sqlite3.h] && echo NSS_USE_SYSTEM_SQLITE=1):` This tests if sqlite is installed and if so it echos the option `NSS_USE_SYSTEM_SQLITE=1` to `make` so that `libsoftokn3.so` will link against the system version of sqlite.

`NSS_DISABLE_GTESTS=1:` If you don't need to run NSS test suite, append this option to `make` command, to prevent the compilation of tests and save some build time.

Configuring NSS

If [p11-kit-0.25.5](#) is installed, the p11-kit trust module (`/usr/lib/pkcs11/p11-kit-trust.so`) can be used as a drop-in replacement for `/usr/lib/libnssckbi.so` to transparently make the system CAs available to NSS aware applications, rather than the static library provided by `/usr/lib/libnssckbi.so`. As the `root` user, execute the following command:

```
ln -sfv ./pkcs11/p11-kit-trust.so /usr/lib/libnssckbi.so
```

Additionally, for dependent applications that do not use the internal database (`/usr/lib/libnssckbi.so`), the `/usr/sbin/make-ca` script included on the [make-ca-1.14](#) page can generate a system wide NSS DB with the `-n` switch, or by modifying the `/etc/make-ca/make-ca.conf` file.

Contents

Installed Programs: certutil, nss-config, and pk12util

Installed Libraries: libcrmfa.a, libfreebl3.so, libfreeblpriv3.so, libnss3.so, libnssckbi.so, libnssckbi-testlib.so, libnssdbm3.so, libnsssysinit.so, libnssutil3.so, libpkcs11testmodule.so, libsmime3.so, libsoftokn3.so, and libssl3.so

Installed Directories: /usr/include/nss

Short Descriptions

<code>certutil</code>	is the Mozilla Certificate Database Tool. It is a command-line utility that can create and modify the Netscape Communicator cert8.db and key3.db database files. It can also list, generate, modify, or delete certificates within the cert8.db file and create or change the password, generate new public and private key pairs, display the contents of the key database, or delete key pairs within the key3.db file
<code>nss-</code> <code>config</code>	is used to determine the NSS library settings of the installed NSS libraries
<code>pk12util</code>	is a tool for importing certificates and keys from pkcs #12 files into NSS or exporting them. It can also list certificates and keys in such files

OpenSSH-9.8p1

Introduction to OpenSSH

The OpenSSH package contains `ssh` clients and the `sshd` daemon. This is useful for encrypting authentication and subsequent traffic over a network. The `ssh` and `scp` commands are secure implementations of `telnet` and `rcp` respectively.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.openbsd.org/pub/OpenBSD/OpenSSH/portable/openssh-9.8p1.tar.gz>
- Download MD5 sum: bc04ff77796758c0b37bd0bc9314cd3f
- Download size: 1.8 MB
- Estimated disk space required: 44 MB (add 22 MB for tests)
- Estimated build time: 0.3 SBU (Using parallelism=4; running the tests takes about 20 minutes, irrespective of processor speed)

OpenSSH Dependencies

Optional

[GDB-15.1](#) (for tests), [Linux-PAM-1.6.1](#) (PAM configuration files from [Shadow-4.16.0](#) are used to create openssh ones), [Xorg Applications](#) (or [Xorg build environment](#), see Command Explanations), [MIT Kerberos V5-1.21.3](#), [Which-2.21](#) (for tests), [libedit](#), [LibreSSL Portable](#), [OpenSC](#), and [libsectok](#)

Optional Runtime (Used only to gather entropy)

[Net-tools-2.10](#), and [Sysstat-12.7.6](#)

Installation of OpenSSH

OpenSSH runs as two processes when connecting to other computers. The first process is a privileged process and controls the issuance of privileges as necessary. The second process communicates with the network. Additional installation steps are necessary to set up the proper environment, which are performed by issuing the following commands as the `root` user:

```
install -v -g sys -m700 -d /var/lib/sshd &&

groupadd -g 50 sshd      &&
useradd -c 'sshd PrivSep' \
-d /var/lib/sshd \
-g sshd \
-s /bin/false \
-u 50 sshd
```

Install OpenSSH by running the following commands:

```
./configure --prefix=/usr \
--sysconfdir=/etc/ssh \
--with-privsep-path=/var/lib/sshd \
--with-default-path=/usr/bin \
--with-superuser-path=/usr/sbin:/usr/bin \
--with-pid-dir=/run      &&
make
```

To test the results, issue: `make -j1 tests`.

Now, as the `root` user:

```
make install &&
install -v -m755 contrib/ssh-copy-id /usr/bin      &&
install -v -m644 contrib/ssh-copy-id.1 \
/usr/share/man/man1      &&
install -v -m755 -d /usr/share/doc/openssh-9.8p1   &&
install -v -m644 INSTALL LICENCE OVERVIEW README* \
/usr/share/doc/openssh-9.8p1
```

Command Explanations

`--sysconfdir=/etc/ssh`: This prevents the configuration files from being installed in `/usr/etc`.

`--with-default-path=/usr/bin` and `--with-superuser-path=/usr/sbin:/usr/bin`: These set `PATH` consistent with LFS and BLFS Shadow package.

`--with-pid-dir=/run`: This prevents OpenSSH from referring to deprecated `/var/run`.

`--with-pam`: This parameter enables Linux-PAM support in the build.

`--with-xauth=$XORG_PREFIX/bin/xauth`: Set the default location for the `xauth` binary for X authentication. The environment variable `XORG_PREFIX` should be set following [Xorg build environment](#). This can also be controlled from `sshd_config` with the `XAuthLocation` keyword. You can omit this switch if Xorg is already installed.

`--with-kerberos5=/usr`: This option is used to include Kerberos 5 support in the build.

`--with-libedit`: This option enables line editing and history features for `sftp`.

Configuring OpenSSH

Config Files

`~/.ssh/*, /etc/ssh/ssh_config, and /etc/ssh/sshd_config`

There are no required changes to any of these files. However, you may wish to view the `/etc/ssh/` files and make any changes appropriate for the security of your system. One recommended change is that you disable `root` login via `ssh`. Execute the following command as the `root` user to disable `root` login via `ssh`:

```
echo "PermitRootLogin no" >> /etc/ssh/sshd_config
```

If you want to be able to log in without typing in your password, first create `~/.ssh/id_rsa` and `~/.ssh/id_rsa.pub` with `ssh-keygen` and then copy `~/.ssh/id_rsa.pub` to `~/.ssh/authorized_keys` on the remote computer that you want to log into. You'll need to change `REMOTE_USERNAME` and `REMOTE_HOSTNAME` for the username and hostname of the remote computer and you'll also need to enter your password for the `ssh-copy-id` command to succeed:

```
ssh-keygen &&
ssh-copy-id -i ~/.ssh/id_ed25519.pub REMOTE_USERNAME@REMOTE_HOSTNAME
```

Once you've got passwordless logins working it's actually more secure than logging in with a password (as the private key is much longer than most people's passwords). If you would like to now disable password logins, as the `root` user:

```
echo "PasswordAuthentication no" >> /etc/ssh/sshd_config &&
echo "KbdInteractiveAuthentication no" >> /etc/ssh/sshd_config
```

If you added Linux-PAM support and you want ssh to use it then you will need to add a configuration file for sshd and enable use of LinuxPAM. Note, ssh only uses PAM to check passwords, if you've disabled password logins these commands are not needed. If you want to use PAM, issue the following commands as the `root` user:

```
sed 's@d/login@d/sshd@g' /etc/pam.d/login > /etc/pam.d/sshd &&
chmod 644 /etc/pam.d/sshd &&
echo "UsePAM yes" >> /etc/ssh/sshd_config
```

Additional configuration information can be found in the man pages for `sshd`, `ssh` and `ssh-agent`.

Systemd Unit

To start the SSH server at system boot, install the `sshd.service` unit included in the [blfs-systemd-units-20240801](#) package.

Note

Changing the setting of `ListenAddress` in `/etc/ssh/sshd_config` is unsupported with the BLFS `sshd` `systemd` unit.

```
make install-sshd
```

Contents

Installed Programs: `scp`, `sftp`, `ssh`, `ssh-add`, `ssh-agent`, `ssh-copy-id`, `ssh-keygen`, `ssh-keyscan`, and `sshd`

Installed Libraries: None

Installed Directories: `/etc/ssh`, `/usr/share/doc/openssh-9.8p1`, and `/var/lib/sshd`

Short Descriptions

<code>scp</code>	is a file copy program that acts like <code>rcp</code> except it uses an encrypted protocol
<code>sftp</code>	is an FTP-like program that works over the SSH1 and SSH2 protocols
<code>ssh</code>	is an <code>rlogin</code> / <code>rsh</code> -like client program except it uses an encrypted protocol
<code>sshd</code>	is a daemon that listens for <code>ssh</code> login requests
<code>ssh-add</code>	is a tool which adds keys to the <code>ssh-agent</code>
<code>ssh-agent</code>	is an authentication agent that can store private keys
<code>ssh-copy-id</code>	is a script that enables logins on remote machines using local keys
<code>ssh-keygen</code>	is a key generation tool
<code>ssh-keyscan</code>	is a utility for gathering public host keys from a number of hosts

p11-kit-0.25.5

Introduction to p11-kit

The p11-kit package provides a way to load and enumerate PKCS #11 (a Cryptographic Token Interface Standard) modules.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/p11-glue/p11-kit/releases/download/0.25.5/p11-kit-0.25.5.tar.xz>
- Download MD5 sum: e9c5675508fcdb8e54aa4c8cb8e794fc
- Download size: 980 KB
- Estimated disk space required: 94 MB (with tests)
- Estimated build time: 0.7 SBU (with tests)

p11-kit Dependencies

Recommended

[libtasn1-4.19.0](#)

Recommended (runtime)

[make-ca-1.14](#)

Optional

[GTK-Doc-1.34.0](#), [libxslt-1.1.42](#), and [nss-3.103](#) (runtime)

Installation of p11-kit

Prepare the distribution specific anchor hook:

```
sed '20,$ d' -i trust/trust-extract-compat &&

cat >> trust/trust-extract-compat << "EOF"
# Copy existing anchor modifications to /etc/ssl/local
/usr/libexec/make-ca/copy-trust-modifications

# Update trust stores
/usr/sbin/make-ca -r
EOF
```

Install p11-kit by running the following commands:

```
mkdir p11-build &&
cd p11-build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D trust_paths=/etc/pki/anchors &&
ninja
```

To test the results, issue: `LC_ALL=C ninja test`.

Now, as the `root` user:

```
ninja install &&
ln -sfv /usr/libexec/p11-kit/trust-extract-compat \
    /usr/bin/update-ca-certificates
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D trust_paths=/etc/pki/anchors`: this switch sets the location of trusted certificates used by libp11-kit.so.

`-D hash_impl=freebl`: Use this switch if you want to use the Freebl library from NSS for SHA1 and MD5 hashing.

`-D gtk_doc=true`: Use this switch if you have installed [GTK-Doc-1.34.0](#) and [libxslt-1.1.42](#) and wish to rebuild the documentation and generate manual pages.

Configuring p11-kit

The p11-kit trust module (`/usr/lib/pkcs11/p11-kit-trust.so`) can be used as a drop-in replacement for `/usr/lib/libnssckbi.so` to transparently make the system CAs available to NSS aware applications, rather than the static list provided by `/usr/lib/libnssckbi.so`. As the `root` user, execute the following commands:

```
ln -sfv ./pkcs11/p11-kit-trust.so /usr/lib/libnssckbi.so
```

Contents

Installed Programs: p11-kit, trust, and update-ca-certificates

Installed Libraries: libp11-kit.so and p11-kit-proxy.so

Installed Directories: /etc/pkcs11, /usr/include/p11-kit-1, /usr/lib/pkcs11, /usr/libexec/p11-kit, /usr/share/gtk-doc/html/p11-kit, and /usr/share/p11-kit

Short Descriptions

p11-kit	is a command line tool that can be used to perform operations on PKCS#11 modules configured on the system
trust	is a command line tool to examine and modify the shared trust policy store
update-ca-certificates	is a command line tool to both extract local certificates from an updated anchor store, and regenerate all anchors and certificate stores on the system. This is done unconditionally on BLFS using the <code>--force</code> and <code>--get</code> flags to <code>make-ca</code> and should likely not be used for automated updates
libp11-kit.so	contains functions used to coordinate initialization and finalization of any PKCS#11 module
p11-kit-proxy.so	is the PKCS#11 proxy module

Polkit-125

Introduction to Polkit

Polkit is a toolkit for defining and handling authorizations. It is used for allowing unprivileged processes to communicate with privileged processes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/polkit-org/polkit/archive/125/polkit-125.tar.gz>
- Download MD5 sum: 8e9f2377fc7b4010bd29b97d2e288b4f
- Download size: 444 KB
- Estimated disk space required: 7.2 MB (with tests)
- Estimated build time: 0.3 SBU (with tests; using parallelism=4)

Polkit Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended)

Recommended

[duktape-2.7.0](#), [libxslt-1.1.42](#), and [Linux-PAM-1.6.1](#)

Note

Since `systemd-logind` uses PAM to register user sessions, it is a good idea to build Polkit with PAM support so `systemd-logind` can track Polkit sessions.

Optional

[GTK-Doc-1.34.0](#), [dbusmock-0.32.1](#) (required for tests), and [SpiderMonkey from Firefox-115.14.0](#) (can be used in place of duktape)

Required Runtime Dependencies

[Systemd-256.4](#)

Optional Runtime Dependencies

One polkit authentication agent for using polkit in the graphical environment: polkit-kde-agent in [Plasma-6.1.4](#) for KDE, the agent built in [gnome-shell-46.4](#) for GNOME3, [polkit-gnome-0.105](#) for XFCE, and [lxqt-policykit-2.0.0](#) for LXQt

Note

If [libxslt-1.1.42](#) is installed, then [docbook-xml-4.5](#) and [docbook-xsl-nons-1.79.2](#) are required. If you have installed [libxslt-1.1.42](#), but you do not want to install any of the DocBook packages mentioned, you will need to use `-D man=false` in the instructions below.

Kernel Configuration

Some tests need user namespace. If running the test suite, enable it:

```
General setup --->
  *-> Namespaces support --->
    [*] User namespace
                                [NAMESPACES]
                                [USER_NS]
```

Installation of Polkit

There should be a dedicated user and group to take control of the `polkitd` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -fg 27 polkitd &&
useradd -c "PolicyKit Daemon Owner" -d /etc/polkit-1 -u 27 \
  -g polkitd -s /bin/false polkitd
```

Install Polkit by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
  --prefix=/usr \
  --buildtype=release \
  -D man=true \
  -D session_tracking=logind \
  -D tests=true
```

Build the package:

```
ninja
```

To test the results, first ensure that the system D-Bus daemon is running, and both [D-Bus Python-1.3.2](#) and [dbusmock-0.32.1](#) are installed. Then run `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D tests=true`: This switch allows to run the test suite of this package. As Polkit is used for authorizations, its integrity can affect system security. So it's recommended to run the test suite building this package.

`-D js_engine=mozjs`: This switch allows using the [SpiderMonkey from Firefox-115.14.0](#) JavaScript engine instead of the [duktape-2.7.0](#) JavaScript engine.

`-D os_type=lfs`: Use this switch if you did not create the `/etc/lfs-release` file or distribution auto detection will fail and you will be unable to use Polkit.

`-D authfw=shadow`: This switch enables the package to use the Shadow rather than the Linux PAM Authentication framework. Use it if you have not installed Linux PAM.

`-D introspection=false`: Use this option if you are certain that you do not need gobject-introspection files for polkit, or do not have installed [GLib-2.80.4](#) with GObject Introspection.

`-D man=false`: Use this option to disable generating and installing manual pages. This is useful if libxslt is not installed.

`-D examples=true`: Use this option to build the example programs.

`-D gtk_doc=true`: Use this option to enable building and installing the API documentation.

Contents

Installed Programs: pkaction, pkcheck, pkexec, pktyagent, and polkitd

Installed Libraries: libpolkit-agent-1.so and libpolkit-gobject-1.so

Installed Directories: /etc/polkit-1, /usr/include/polkit-1, /usr/lib/polkit-1, /usr/share/gtk-doc/html/polkit-1, and /usr/share/polkit-1

Short Descriptions

<code>pkaction</code>	is used to obtain information about registered PolicyKit actions
<code>pkcheck</code>	is used to check whether a process is authorized for action
<code>pkexec</code>	allows an authorized user to execute a command as another user
<code>pktyagent</code>	is used to start a textual authentication agent for the subject
<code>polkitd</code>	provides the org.freedesktop.PolicyKit1 D-Bus service on the system message bus
<code>libpolkit-agent-1.so</code>	contains the Polkit authentication agent API functions
<code>libpolkit-gobject-1.so</code>	contains the Polkit authorization API functions

polkit-gnome-0.105

Introduction to Polkit GNOME

The Polkit GNOME package provides an Authentication Agent for Polkit that integrates well with the GNOME Desktop environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/polkit-gnome/0.105/polkit-gnome-0.105.tar.xz>
- Download MD5 sum: 50ecad37c8342fb4a52f590db7530621
- Download size: 305 KB
- Estimated disk space required: 5.0 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/polkit-gnome-0.105-consolidated_fixes-1.patch

Polkit GNOME Dependencies

Required

[AccountsService-23.13.9](#), [GTK+-3.24.43](#), and [Polkit-125](#)

Installation of Polkit GNOME

First, apply some fixes that allow for the proper user icon to be used, as well as some security fixes:

```
patch -Np1 -i ../polkit-gnome-0.105-consolidated_fixes-1.patch
```

Install Polkit GNOME by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Configuring Polkit GNOME

Automatic Startup

For the authentication framework to work, `polkit-gnome-authentication-agent-1` needs to be started. However, `make install` did not install a startup file for the Polkit GNOME so you have to create it by yourself.

Issue the following commands as the `root` user to create a startup file for Polkit GNOME:

```
mkdir -p /etc/xdg/autostart &&
cat > /etc/xdg/autostart/polkit-gnome-authentication-agent-1.desktop << "EOF"
[Desktop Entry]
Name=PolicyKit Authentication Agent
Comment=PolicyKit Authentication Agent
Exec=/usr/libexec/polkit-gnome-authentication-agent-1
Terminal=false
Type=Application
Categories=
NoDisplay=true
OnlyShowIn=GNOME;XFCE;Unity;
AutostartCondition=GNOME3 unless-session gnome
EOF
```

Contents

Installed Program: `polkit-gnome-authentication-agent-1`

Installed Libraries: None

Installed Directory: None

Short Descriptions

`polkit-gnome-authentication-agent-1` is the Polkit authentication agent

Shadow-4.16.0

Introduction to Shadow

Shadow was indeed installed in LFS and there is no reason to reinstall it unless you installed CrackLib or Linux-PAM after your LFS system was completed. If you have installed CrackLib after LFS, then reinstalling Shadow will enable strong password support. If you have installed Linux-PAM, reinstalling Shadow will allow programs such as `login` and `su` to utilize PAM.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/shadow-maint/shadow/releases/download/4.16.0/shadow-4.16.0.tar.xz>
- Download MD5 sum: eb70bad3316d08f0d3bb3d4bbeccb3b4
- Download size: 2.1 MB
- Estimated disk space required: 100 MB
- Estimated build time: 0.2 SBU

Shadow Dependencies

Required

[Linux-PAM-1.6.1](#) or [CrackLib-2.10.2](#)

Optional

[libbsd](#) and [tcb](#)

Installation of Shadow

Important

The installation commands shown below are for installations where Linux-PAM has been installed and Shadow is being reinstalled to support the Linux-PAM installation.

If you are reinstalling Shadow to provide strong password support using the CrackLib library without using Linux-PAM, ensure you add the `--with-libcrack` parameter to the `configure` script below and also issue the following command:

```
sed -i 's@DICTPATH.*@DICTPATH\t/lib/cracklib/pw_dict@' etc/login.defs
```

Warning

If reinstalling shadow for a version update, be sure to reaccomplish the Linux-PAM configuration below. The installation of shadow overwrites many of the files in `/etc/pam.d/`.

Reinstall Shadow by running the following commands:

```
sed -i 's/groups$(EXEEXT) //' src/Makefile.in      &&
find man -name Makefile.in -exec sed -i 's/groups\.\.1 / /' {} \; &&
find man -name Makefile.in -exec sed -i 's/getspnam\.\.3 / /' {} \; &&
find man -name Makefile.in -exec sed -i 's/passwd\.\.5 / /' {} \; &&

sed -e 's@#ENCRYPT_METHOD DES@ENCRYPT_METHOD YESCRYPT@' \
-e 's@/var/spool/mail@/var/mail@' \
-e '/PATH=/{}@sbin:@@;bin:@@' \
-i etc/login.defs      &&

./configure --sysconfdir=/etc \
--disable-static \
--without-libbsd \
--with-{b,yes}crypt &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make exec_prefix=/usr pamddir= install
```

The man pages were installed in LFS, but if reinstallation is desired, run (as the `root` user):

```
make -C man install-man
```

Command Explanations

`sed -i 's/groups$(EXEEXT) //' src/Makefile.in`: This sed is used to suppress the installation of the `groups` program as the version from the Coreutils package installed during LFS is preferred.

`find man -name Makefile.in -exec ... {} \;`: The first command is used to suppress the installation of the `groups` man pages so the existing ones installed from the Coreutils package are not replaced. The two other commands prevent installation of manual pages that are already installed by Man-pages in LFS.

`sed -e 's@#ENCRYPT_METHOD DES@ENCRYPT_METHOD YESCRYPT@' -e 's@/var/spool/mail@/var/mail@' -e '/PATH=/{}@sbin:@@;bin:@@' -i etc/login.defs`: Instead of using the default 'DES' method, this command modifies the installation to use the much more secure 'YESCRYPT' method of hashing passwords, which also allows passwords longer than eight characters. The command

also changes the obsolete `/var/spool/mail` location for user mailboxes that Shadow uses by default to the `/var/mail` location. It also changes the default path to be consistent with that set in LFS.

`--without-libbsd`: Prevents looking for the `readpassphrase` function, which can be found only in `libbsd`, which we do not have in BLFS. An internal implementation of `readpassphrase` is used instead.

`pamddir=`: Prevents installation of the shipped PAM configuration files into `/etc/pam.d`. The shipped configuration does not work with the BLFS PAM configuration and we will create these configuration files explicitly.

Configuring Linux-PAM to Work with Shadow

Note

The rest of this page is devoted to configuring Shadow to work properly with Linux-PAM. If you do not have Linux-PAM installed, and you reinstalled Shadow to support strong passwords via the CrackLib library, no further configuration is required.

Config Files

`/etc/pam.d/*` or alternatively `/etc/pam.conf`, `/etc/login.defs` and `/etc/security/*`

Configuration Information

Configuring your system to use Linux-PAM can be a complex task. The information below will provide a basic setup so that Shadow's login and password functionality will work effectively with Linux-PAM. Review the information and links on the [Linux-PAM-1.6.1](#) page for further configuration information. For information specific to integrating Shadow, Linux-PAM and libpwquality, you can visit the following link:

- https://deer-run.com/users/hal/linux_passwords_pam.html

Configuring `/etc/login.defs`

The `login` program currently performs many functions which Linux-PAM modules should now handle. The following `sed` command will comment out the appropriate lines in `/etc/login.defs`, and stop `login` from performing these functions (a backup file named `/etc/login.defs.orig` is also created to preserve the original file's contents). Issue the following commands as the `root` user:

```
install -v -m644 /etc/login.defs /etc/login.defs.orig &&
for FUNCTION in FAIL_DELAY \
    FAILLOG_ENAB \
    LASTLOG_ENAB \
    MAIL_CHECK_ENAB \
    OBSCURE_CHECKS_ENAB \
    PORTTIME_CHECKS_ENAB \
    QUOTAS_ENAB \
    CONSOLE_MOTD_FILE \
    FTMP_FILE_NOLOGINS_FILE \
    ENV_HZ_PASS_MIN_LEN \
    SU_WHEEL_ONLY \
    CRACKLIB_DICTPATH \
    PASS_CHANGE_TRIES \
    PASS_ALWAYS_WARN \
    CHFN_AUTH_ENCRYPT_METHOD \
    ENVIRON_FILE
do
    sed -i "s/^${FUNCTION}/# &/" /etc/login.defs
done
```

Configuring the `/etc/pam.d/` Files

As mentioned previously in the Linux-PAM instructions, Linux-PAM has two supported methods for configuration. The commands below assume that you've chosen to use a directory based configuration, where each program has its own configuration file. You can optionally use a single `/etc/pam.conf` configuration file by using the text from the files below, and supplying the program name as an additional first field for each line.

As the `root` user, create the following Linux-PAM configuration files in the `/etc/pam.d/` directory (or add the contents to the `/etc/pam.conf` file) using the following commands:

'login'

```
cat > /etc/pam.d/login << "EOF"
# Begin /etc/pam.d/login

# Set failure delay before next prompt to 3 seconds
auth optional pam_faildelay.so delay=3000000

# Check to make sure that the user is allowed to login
auth requisite pam_nologin.so

# Check to make sure that root is allowed to login
# Disabled by default. You will need to create /etc/securetty
# file for this module to function. See man 5 securetty.
#auth required pam_securetty.so

# Additional group memberships - disabled by default
#auth optional pam_group.so

# include system auth settings
auth include system-auth

# check access for the user
account required pam_access.so

# include system account settings
account include system-account

# Set default environment variables for the user
session required pam_env.so

# Set resource limits for the user
session required pam_limits.so

# Display the message of the day - Disabled by default
#session optional pam_motd.so

# Check user's mail - Disabled by default
#session optional pam_mail.so      standard quiet

# include system session and password settings
session include system-session
password include system-password

# End /etc/pam.d/login
EOF
```

'passwd'

```
cat > /etc/pam.d/passwd << "EOF"
# Begin /etc/pam.d/passwd

password include system-password

# End /etc/pam.d/passwd
EOF
```

'su'

```
cat > /etc/pam.d/su << "EOF"
# Begin /etc/pam.d/su

# always allow root
auth sufficient pam_rootok.so

# Allow users in the wheel group to execute su without a password
# disabled by default
#auth sufficient pam_wheel.so trust use_uid

# include system auth settings
auth include system-auth

# limit su to users in the wheel group
```

```

# disabled by default
auth      required    pam_wheel.so use_uid

# include system account settings
account   include     system-account

# Set default environment variables for the service user
session   required    pam_env.so

# include system session settings
session   include     system-session

# End /etc/pam.d/su
EOF

```

'chpasswd' and 'newusers'

```

cat > /etc/pam.d/chpasswd << "EOF"
# Begin /etc/pam.d/chpasswd

# always allow root
auth      sufficient  pam_rootok.so

# include system auth and account settings
auth      include     system-auth
account   include     system-account
password  include     system-password

# End /etc/pam.d/chpasswd
EOF

sed -e s/chpasswd/newusers/ /etc/pam.d/chpasswd >/etc/pam.d/newusers

```

'chage'

```

cat > /etc/pam.d/chage << "EOF"
# Begin /etc/pam.d/chage

# always allow root
auth      sufficient  pam_rootok.so

# include system auth and account settings
auth      include     system-auth
account   include     system-account

# End /etc/pam.d/chage
EOF

```

Other shadow utilities

```

for PROGRAM in chfn chgpasswd chsh groupadd groupdel \
              groupmems groupmod useradd userdel usermod
do
  install -v -m644 /etc/pam.d/chage /etc/pam.d/${PROGRAM}
  sed -i "s/chage/${PROGRAM}/" /etc/pam.d/${PROGRAM}
done

```

Warning

At this point, you should do a simple test to see if Shadow is working as expected. Open another terminal and log in as `root`, and then run `login` and login as another user. If you do not see any errors, then all is well and you should proceed with the rest of the configuration. If you did receive errors, stop now and double check the above configuration files manually. Any error is the sign of an error in the above procedure. You can also run the test suite from the Linux-PAM package to assist you in determining the problem. If you cannot find and fix the error, you should recompile Shadow adding the `--without-libpam` switch to the `configure` command in the above instructions (also move the `/etc/login.defs.orig` backup file to `/etc/login.defs`). If you fail to do this and the errors remain, you will be unable to log into your system.

Configuring Login Access

Instead of using the `/etc/login.access` file for controlling access to the system, Linux-PAM uses the `pam_access.so` module along with the `/etc/security/access.conf` file. Rename the `/etc/login.access` file using the following command:

```
if [ -f /etc/login.access ]; then mv -v /etc/login.access{,.NOUSER}; fi
```

Configuring Resource Limits

Instead of using the `/etc/limits` file for limiting usage of system resources, Linux-PAM uses the `pam_limits.so` module along with the `/etc/security/limits.conf` file. Rename the `/etc/limits` file using the following command:

```
if [ -f /etc/limits ]; then mv -v /etc/limits{,.NOUSER}; fi
```

Caution

Be sure to test the login capabilities of the system before logging out. Errors in the configuration can cause a permanent lockout requiring a boot from an external source to correct the problem.

Contents

A list of the installed files, along with their short descriptions can be found at [..../..../..../lfs/view/12.2-systemd/chapter08/shadow.html#contents-shadow](#).

ssh-askpass-9.8p1

Introduction to ssh-askpass

The `ssh-askpass` is a generic executable name for many packages, with similar names, that provide a interactive X service to grab password for packages requiring administrative privileges to be run. It prompts the user with a window box where the necessary password can be inserted. Here, we choose Damien Miller's package distributed in the OpenSSH tarball.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.openbsd.org/pub/OpenBSD/OpenSSH/portable/openssh-9.8p1.tar.gz>
- Download MD5 sum: bc04ff77796758c0b37bd0bc9314cd3f
- Download size: 1.8 MB
- Estimated disk space required: 10 MB
- Estimated build time: less than 0.1 SBU

ssh-askpass Dependencies

Required

[GTK+-3.24.43](#), [Sudo-1.9.15p5](#) (runtime), [Xorg Libraries](#), and [a graphical environment](#) (runtime)

Installation of ssh-askpass

Install `ssh-askpass` by running the following commands:

```
cd contrib &&
make gnome-ssh-askpass3
```

Now, as the `root` user:

```
install -v -d -m755          /usr/libexec/openssh/contrib  &&
install -v -m755    gnome-ssh-askpass3 /usr/libexec/openssh/contrib  &&
ln -sv -f contrib/gnome-ssh-askpass3  /usr/libexec/openssh/ssh-askpass
```

The use of /usr/libexec.openssh/contrib and a symlink is justified by the eventual necessity of a different program for that service.

Configuring ssh-askpass

Configuration Information

As the `root` user, configure [Sudo-1.9.15p5](#) to use ssh-askpass:

```
cat >> /etc/sudo.conf << "EOF" &&
# Path to askpass helper program
Path askpass /usr/libexec.openssh/ssh-askpass
EOF
chmod -v 0644 /etc/sudo.conf
```

If a given graphical <application> requires administrative privileges, use `sudo -A <application>` from an x-terminal, from a Window Manager menu and/or replace "Exec=<application> ..." by "Exec=sudo -A <application> ..." in the <application>.desktop file.

Contents

Installed Programs: gnome-ssh-askpass3, ssh-askpass (symlink to `gnome-ssh-askpass3`)

Installed Library: None

Installed Directory: /usr/libexec.openssh/contrib

stunnel-5.72

Introduction to stunnel

The stunnel package contains a program that allows you to encrypt arbitrary TCP connections inside SSL (Secure Sockets Layer) so you can easily communicate with clients over secure channels. stunnel can also be used to tunnel PPP over network sockets without changes to the server package source code.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.stunnel.org/downloads/archive/5.x/stunnel-5.72.tar.gz>
- Download MD5 sum: 61d35d2df1b96b23ed6fb20cf5ddb613
- Download size: 880 KB
- Estimated disk space required: 7.5 MB
- Estimated build time: less than 0.1 SBU

stunnel Dependencies

Optional

[libnsl-2.0.1](#), [netcat](#) (required for tests), [tcpwrappers](#), and [TOR](#)

Installation of stunnel

The `stunnel` daemon will be run in a `chroot` jail by an unprivileged user. Create the new user and group using the following commands as the `root` user:

```
groupadd -g 51 stunnel &&
useradd -c "stunnel Daemon" -d /var/lib/stunnel \
-g stunnel -s /bin/false -u 51 stunnel
```

Note

A signed SSL Certificate and a Private Key is necessary to run the `stunnel` daemon. After the package is installed, there are instructions to generate them. However, if you own or have already created a signed SSL

Certificate you wish to use, copy it to `/etc/stunnel/stunnel.pem` before starting the build (ensure only `root` has read and write access). The `.pem` file must be formatted as shown below:

```
-----BEGIN PRIVATE KEY-----  
<many encrypted lines of private key>  
-----END PRIVATE KEY-----  
-----BEGIN CERTIFICATE-----  
<many encrypted lines of certificate>  
-----END CERTIFICATE-----  
-----BEGIN DH PARAMETERS-----  
<encrypted lines of dh parms>  
-----END DH PARAMETERS-----
```

Install stunnel by running the following commands:

```
./configure --prefix=/usr      \  
          --sysconfdir=/etc    \  
          --localstatedir=/var &&  
make
```

If you have installed the optional netcat application, the regression tests can be run with `make check`.

Now, as the `root` user:

```
make docdir=/usr/share/doc/stunnel-5.72 install
```

Install the included systemd unit by running the following command as the `root` user:

```
install -v -m644 tools/stunnel.service /usr/lib/systemd/system
```

If you do not already have a signed SSL Certificate and Private Key, create the `stunnel.pem` file in the `/etc/stunnel` directory using the command below. You will be prompted to enter the necessary information. Ensure you reply to the

```
Common Name (FQDN of your server) [localhost]:
```

prompt with the name or IP address you will be using to access the service(s).

To generate a certificate, as the `root` user, issue:

```
make cert
```

Command Explanations

`make docdir=... install`: This command installs the package and changes the documentation installation directory to standard naming conventions.

Configuring stunnel

Config Files

`/etc/stunnel/stunnel.conf`

Configuration Information

As the `root` user, create the directory used for the `.pid` file created when the stunnel daemon starts:

```
install -v -m750 -o stunnel -g stunnel -d /var/lib/stunnel/run &&  
chown stunnel:stunnel /var/lib/stunnel
```

Next, create a basic `/etc/stunnel/stunnel.conf` configuration file using the following commands as the `root` user:

```
cat > /etc/stunnel/stunnel.conf << "EOF"  
; File: /etc/stunnel/stunnel.conf  
  
; Note: The pid and output locations are relative to the chroot location.  
  
pid    = /run/stunnel.pid  
chroot = /var/lib/stunnel  
client = no
```

```

setuid = stunnel
setgid = stunnel
cert   = /etc/stunnel/stunnel.pem

;debug = 7
;output = stunnel.log

;[https]
;accept  = 443
;connect = 80
;; "TIMEOUTclose = 0" is a workaround for a design flaw in Microsoft SSL
;; Microsoft implementations do not use SSL close-notify alert and thus
;; they are vulnerable to truncation attacks
;TIMEOUTclose = 0

EOF

```

Finally, add the service(s) you wish to encrypt to the configuration file. The format is as follows:

```

[<service>]
accept  = <hostname:portnumber>
connect = <hostname:portnumber>

```

For a full explanation of the commands and syntax used in the configuration file, issue `man stunnel`.

Systemd Unit

To start the `stunnel` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable stunnel
```

Contents

Installed Programs: stunnel and stunnel3

Installed Library: libstunnel.so

Installed Directories: /{etc,lib,var/lib}/stunnel and /usr/share/doc/stunnel-5.72

Short Descriptions

<code>stunnel</code>	is a program designed to work as an SSL encryption wrapper between remote clients and local or remote servers
<code>stunnel3</code>	is a Perl wrapper script to use <code>stunnel</code> 3.x syntax with <code>stunnel</code> 4.05 or later
<code>libstunnel.so</code>	contains the API functions required by <code>stunnel</code>

Sudo-1.9.15p5

Introduction to Sudo

The Sudo package allows a system administrator to give certain users (or groups of users) the ability to run some (or all) commands as `root` or another user while logging the commands and arguments.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.sudo.ws/dist/sudo-1.9.15p5.tar.gz>
- Download MD5 sum: 4166279cb188ecb6641c7a2ba5f68270
- Download size: 5.1 MB
- Estimated disk space required: 53 MB (add 18 MB for tests)
- Estimated build time: 0.2 SBU (with parallelism=4; add 0.1 SBU for tests)

Sudo Dependencies

Optional

[Linux-PAM-1.6.1](#), [MIT Kerberos V5-1.21.3](#), [OpenLDAP-2.6.8](#), [MTA](#) (that provides a `sendmail` command), [AFS](#), [libaudit](#), [Opie](#), and [Sssd](#)

Installation of Sudo

Install Sudo by running the following commands:

```
./configure --prefix=/usr          \
            --libexecdir=/usr/lib \
            --with-secure-path   \
            --with-env-editor    \
            --docdir=/usr/share/doc/sudo-1.9.15p5 \
            --with-passprompt="[sudo] password for %p: " &&
make
```

To test the results, issue: `env LC_ALL=C make check |& tee make-check.log`. Check the results with `grep failed make-check.log`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--libexecdir=/usr/lib`: This switch controls where private programs are installed. Everything in that directory is a library, so they belong under `/usr/lib` instead of `/usr/libexec`.

`--with-secure-path`: This switch transparently adds `/sbin` and `/usr/sbin` directories to the `PATH` environment variable.

`--with-env-editor`: This switch enables use of the environment variable `EDITOR` for `visudo`.

`--with-passprompt`: This switch sets the password prompt. The `%p` will be expanded to the name of the user whose password is being requested.

`--without-pam`: This switch avoids building Linux-PAM support when Linux-PAM is installed on the system.

`--with-all-insults`: This switch includes all the sudo insult sets. Insults are printed if the user types a bad password, and if enabled in `/etc/sudoers`. Use `--with-insults` to have them enabled by default. Various sets of insults can be selected with some other switches.

Note

There are many options to sudo's `configure` command. Check the `configure --help` output for a complete list.

Configuring Sudo

Config File

`/etc/sudoers`

Configuration Information

The `sudoers` file can be quite complicated. It is composed of two types of entries: aliases (basically variables) and user specifications (which specify who may run what). The installation installs a default configuration that has no privileges installed for any user.

A couple of common configuration changes are to set the path for the super user and to allow members of the wheel group to execute all commands after providing their own credentials. Use the following commands to create the `/etc/sudoers.d/00-sudo` configuration file as the `root` user:

```
cat > /etc/sudoers.d/00-sudo << "EOF"
Defaults secure_path="/usr/sbin:/usr/bin"
%wheel ALL=(ALL) ALL
EOF
```

Note

In very simple installations where there is only one user, it may be easier to just edit the `/etc/sudoers` file directly. In that case, the `secure_path` entry may not be needed and using `sudo -E ...` can import the non-privileged user's full environment into the privileged session.

The files in the `/etc/sudoers.d` directory are parsed in sorted lexical order. Be careful that entries in an added file do not overwrite previous entries.

For details, see `man sudoers`.

Note

The Sudo developers highly recommend using the `visudo` program to edit the `sudoers` file. This will provide basic sanity checking like syntax parsing and file permission to avoid some possible mistakes that could lead to a vulnerable configuration.

If PAM is installed on the system, Sudo is built with PAM support. In that case, issue the following command as the `root` user to create the PAM configuration file:

```
cat > /etc/pam.d/sudo << "EOF"
# Begin /etc/pam.d/sudo

# include the default auth settings
auth      include      system-auth

# include the default account settings
account   include      system-account

# Set default environment variables for the service user
session   required     pam_env.so

# include system session defaults
session   include      system-session

# End /etc/pam.d/sudo
EOF
chmod 644 /etc/pam.d/sudo
```

Contents

Installed Programs: cvtsudoers, sudo, sudo_logsrvd, sudo_sendlog, sudoedit (symlink), sudoreplay, and visudo

Installed Libraries: audit_json.so, group_file.so, libsudo_util.so, sudoers.so, sudo_intercept.so, sudo_noexec.so, and system_group.so

Installed Directories: /etc/sudoers.d, /usr/lib/sudo, /usr/share/doc/sudo-1.9.15p5, and /var/lib/sudo

Short Descriptions

<code>cvtsudoers</code>	converts between sudoers file formats
<code>sudo</code>	executes a command as another user as permitted by the <code>/etc/sudoers</code> configuration file
<code>sudo_logsrvd</code>	is a sudo event and I/O log server
<code>sudo_sendlog</code>	sends sudo I/O logs to the log server
<code>sudoedit</code>	is a symlink to <code>sudo</code> that implies the <code>-e</code> option to invoke an editor as another user
<code>sudoreplay</code>	is used to play back or list the output logs created by <code>sudo</code>
<code>visudo</code>	allows for safer editing of the <code>sudoers</code> file

Tripwire-2.4.3.7

Introduction to Tripwire

The Tripwire package contains programs used to verify the integrity of the files on a given system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Tripwire/tripwire-open-source/releases/download/2.4.3.7/tripwire-open-source-2.4.3.7.tar.gz>
- Download MD5 sum: a5cf1bc2f235f5d8ca458f00548db6ee
- Download size: 980 KB
- Estimated disk space required: 29 MB
- Estimated build time: 1.6 SBU (scripting install)

Tripwire Dependencies

Optional

An [MTA](#)

Installation of Tripwire

Compile Tripwire by running the following commands:

```
sed -e '/^CLOBBER/s/false/true/' \
-e 's|TWDB="${prefix}!TWDB="/var|' \
-e '/TWMAN/ s|${prefix}|/usr/share|' \
-e '/TWDODCS/s|${prefix}|/doc/tripwire|/usr/share/doc/tripwire-2.4.3.7|' \
-i installer/install.cfg \
&&

find . -name Makefile.am | xargs \
    sed -i 's/^[[[:alpha:]_]*_HEADERS.*=/noinst_HEADERS =/' \
&&

sed '/dist/d' -i man/man?/Makefile.am \
autoreconf -fi \
&&

./configure --prefix=/usr --sysconfdir=/etc/tripwire \
make CPPFLAGS=-std=c++11
```

Note

The default configuration is to use a local MTA. If you don't have an MTA installed and have no wish to install one, modify `install/install.cfg` to use an SMTP server instead. Otherwise the install will fail.

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
cp -v policy/*.txt /usr/share/doc/tripwire-2.4.3.7
```

Note

During `make install`, several questions are asked, including passwords. If you want to make a script, you have to apply a sed before running `make install`:

```
sed -i -e 's@installer/install.sh@& -n -s <site-password> -l <local-password>@' Makefile
```

Of course, you should do this with dummy passwords and change them later.

Another issue when scripting is that the installer exits when the standard input is not a terminal. You may disable this behavior with the following sed:

```
sed '/-t 0/,+3d' -i installer/install.sh
```

Command Explanations

`sed ... installer/install.cfg`: This command tells the package to install the program database and reports in `/var/lib/tripwire` and sets the proper location for man pages and documentation.

`find ..., sed ..., and autoreconf -fi`: The build system is unusable as is, and has to be modified for the build to succeed.

`CPPFLAGS=-std=c++11`: Setting the C++ preprocessor flags to version 11 is necessary to prevent a conflict with the default version which is c++17 in recent version of gcc.

`make install`: This command creates the Tripwire security keys as well as installing the binaries. There are two keys: a site key and a local key which are stored in `/etc/tripwire/`.

`cp -v policy/*.txt /usr/doc/tripwire-2.4.3.7`: This command installs the tripwire sample policy files with the other tripwire documentation.

Configuring Tripwire

Config Files

`/etc/tripwire/*`

Configuration Information

Tripwire uses a policy file to determine which files are integrity checked. The default policy file (`/etc/tripwire/twpol.txt`) is for a default installation and will need to be updated for your system.

Policy files should be tailored to each individual distribution and/or installation. Some example policy files can be found in `/usr/share/doc/tripwire/`.

If desired, copy the policy file you'd like to try into `/etc/tripwire/` instead of using the default policy file, `twpol.txt`. It is, however, recommended that you edit your policy file. Get ideas from the examples above and read `/usr/share/doc/tripwire/policyguide.txt` for additional information. `twpol.txt` is a good policy file for learning about Tripwire as it will note any changes to the file system and can even be used as an annoying way of keeping track of changes for uninstallation of software.

After your policy file has been edited to your satisfaction you may begin the configuration steps (perform as the `root` user):

```
twadmin --create-polfile --site-keyfile /etc/tripwire/site.key \
    /etc/tripwire/twpol.txt &&
tripwire --init
```

Depending on your system and the contents of the policy file, the initialization phase above can take a relatively long time.

Usage Information

Tripwire will identify file changes in the critical system files specified in the policy file. Using Tripwire while making frequent changes to these directories will flag all these changes. It is most useful after a system has reached a configuration that the user considers stable.

To use Tripwire after creating a policy file to run a report, use the following command:

```
tripwire --check > /etc/tripwire/report.txt
```

View the output to check the integrity of your files. An automatic integrity report can be produced by using a cron facility to schedule the runs.

Reports are stored in binary and, if desired, encrypted. View reports, as the `root` user, with:

```
twprint --print-report -r /var/lib/tripwire/report/<report-name.twr>
```

After you run an integrity check, you should examine the report (or email) and then modify the Tripwire database to reflect the changed files on your system. This is so that Tripwire will not continually notify you that files you intentionally changed are a security violation. To do this you must first `ls -l /var/lib/tripwire/report/` and note the name of the newest file which starts with your system name as presented by the command `uname -n` and ends in `.twr`. These files were created during report creation and the most current one is needed to update the Tripwire database of your system. As the `root` user, type in the following command making the appropriate report name:

```
tripwire --update --twrfile /var/lib/tripwire/report/<report-name.twr>
```

You will be placed into Vim with a copy of the report in front of you. If all the changes were good, then just type `:wq` and after entering your local key, the database will be updated. If there are files which you still want to be warned about, remove the 'x' before the filename in the report and type `:wq`.

Changing the Policy File

If you are unhappy with your policy file and would like to modify it or use a new one, modify the policy file and then execute the following commands as the `root` user:

```
twadmin --create-polfile /etc/tripwire/twpol.txt &&
tripwire --init
```

Contents

Installed Programs: siggen, tripwire, twadmin, and twprint

Installed Libraries: None

Installed Directories: /etc/tripwire, /var/lib/tripwire, and /usr/share/doc/tripwire-2.4.3.7

Short Descriptions

<code>siggen</code>	is a signature gathering utility that displays the hash function values for the specified files
<code>tripwire</code>	is the main file integrity checking program
<code>twadmin</code>	administrative and utility tool used to perform certain administrative functions related to Tripwire files and configuration options
<code>twprint</code>	prints Tripwire database and report files in clear text format

Chapter 5. File Systems and Disk Management

Journaling file systems reduce the time needed to recover a file system that was not unmounted properly. While this can be extremely important in reducing downtime for servers, it has also become popular for desktop environments. This chapter contains other journaling file systems you can use instead of the default LFS extended file system (ext2/3/4). It also provides introductory material on managing disk arrays.

About initramfs

The only purpose of an initramfs is to mount the root filesystem. The initramfs is a complete set of directories that you would find on a normal root filesystem. It is bundled into a single cpio archive and compressed with one of several compression algorithms.

At boot time, the boot loader loads the kernel and the initramfs image into memory and starts the kernel. The kernel checks for the presence of the initramfs and, if found, mounts it as / and runs /init. The init program is typically a shell script. Note that the boot process takes longer, possibly significantly longer, if an initramfs is used.

For most distributions, kernel modules are the biggest reason to have an initramfs. In a general distribution, there are many unknowns such as file system types and disk layouts. In a way, this is the opposite of LFS where the system capabilities and layout are known and a custom kernel is normally built. In this situation, an initramfs is rarely needed.

There are only four primary reasons to have an initramfs in the LFS environment: loading the rootfs from a network, loading it from an LVM logical volume, having an encrypted rootfs where a password is required, or for the convenience of specifying the rootfs as a LABEL or UUID. Anything else usually means that the kernel was not configured properly.

Building an initramfs

If you do decide to build an initramfs, the following scripts will provide a basis to do it. The scripts will allow specifying a rootfs via partition UUID or partition LABEL or a rootfs on an LVM logical volume. They do not support an encrypted root file system or mounting the rootfs over a network card. For a more complete capability see [the LFS Hints](#) or [dracut](#).

To install these scripts, run the following commands as the `root` user:

```
cat > /usr/sbin/mkinitramfs << "EOF"
#!/bin/bash
# This file based in part on the mkinitramfs script for the LFS LiveCD
# written by Alexander E. Patrakov and Jeremy Huntwork.

copy()
{
    local file

    if [ "$2" = "lib" ]; then
        file=$(PATH=/usr/lib type -p $1)
    else
        file=$(type -p $1)
    fi

    if [ -n "$file" ] ; then
        cp $file $WDIR/usr/$2
    else
        echo "Missing required file: $1 for directory $2"
    fi
}
```

```

        rm -rf $WDIR
        exit 1
    fi
}

if [ -z $1 ] ; then
    INITRAMFS_FILE=initrd.img-no-kmods
else
    KERNEL_VERSION=$1
    INITRAMFS_FILE=initrd.img-$KERNEL_VERSION
fi

if [ -n "$KERNEL_VERSION" ] && [ ! -d "/usr/lib/modules/$1" ] ; then
    echo "No modules directory named $1"
    exit 1
fi

printf "Creating $INITRAMFS_FILE... "

binfiles="sh cat cp dd killall ls mkdir mknod mount "
binfiles="$binfiles umount sed sleep ln rm uname"
binfiles="$binfiles readlink basename"

# Systemd installs udevadm in /bin. Other udev implementations have it in /sbin
if [ -x /usr/bin/udevadm ] ; then binfiles="$binfiles udevadm"; fi

sbinfiles="modprobe blkid switch_root"

# Optional files and locations
for f in mdadm mdmon udevd udevadm; do
    if [ -x /usr/sbin/$f ] ; then sbinfiles="$sbinfiles $f"; fi
done

# Add lvm if present (cannot be done with the others because it
# also needs dmsetup
if [ -x /usr/sbin/lvm ] ; then sbinfiles="$sbinfiles lvm dmsetup"; fi

unsorted=$(mktemp /tmp/unsorted.XXXXXXXXXX)

DATADIR=/usr/share/mkinitramfs
INITIN=init.in

# Create a temporary working directory
WDIR=$(mktemp -d /tmp/initrd-work.XXXXXXXXXX)

# Create base directory structure
mkdir -p $WDIR/{dev,run,sys,proc,usr/{bin,lib/{firmware,modules},sbin}}
mkdir -p $WDIR/etc/{modprobe.d,udev/rules.d}
touch $WDIR/etc/modprobe.d/modprobe.conf
ln -s usr/bin $WDIR/bin
ln -s usr/lib $WDIR/lib
ln -s usr/sbin $WDIR/sbin
ln -s lib       $WDIR/lib64

# Create necessary device nodes
mknod -m 640 $WDIR/dev/console c 5 1
mknod -m 664 $WDIR/dev/null   c 1 3

# Install the udev configuration files
if [ -f /etc/udev/udev.conf ]; then
    cp /etc/udev/udev.conf $WDIR/etc/udev/udev.conf
fi

for file in $(find /etc/udev/rules.d/ -type f) ; do
    cp $file $WDIR/etc/udev/rules.d
done

# Install any firmware present
cp -a /usr/lib/firmware $WDIR/usr/lib

# Copy the RAID configuration file if present
if [ -f /etc/mdadm.conf ] ; then
    cp /etc/mdadm.conf $WDIR/etc
fi

# Install the init file

```

```

install -m0755 $DATADIR/$INITIN $WDIR/init

if [ -n "$KERNEL_VERSION" ] ; then
    if [ -x /usr/bin/kmod ] ; then
        binfiles="$binfiles kmod"
    else
        binfiles="$binfiles lsmod"
        sbinfiles="$sbinfiles insmod"
    fi
fi

# Install basic binaries
for f in $binfiles ; do
    ldd /usr/bin/$f | sed "s/\t// " | cut -d " " -f1 >> $unsorted
    copy /usr/bin/$f bin
done

for f in $sbinfiles ; do
    ldd /usr/sbin/$f | sed "s/\t// " | cut -d " " -f1 >> $unsorted
    copy $f sbin
done

# Add udevd libraries if not in /usr/sbin
if [ -x /usr/lib/udev/udevd ] ; then
    ldd /usr/lib/udev/udevd | sed "s/\t// " | cut -d " " -f1 >> $unsorted
elif [ -x /usr/lib/systemd/systemd-udevd ] ; then
    ldd /usr/lib/systemd/systemd-udevd | sed "s/\t// " | cut -d " " -f1 >> $unsorted
fi

# Add module symlinks if appropriate
if [ -n "$KERNEL_VERSION" ] && [ -x /usr/bin/kmod ] ; then
    ln -s kmod $WDIR/usr/bin/lsmod
    ln -s kmod $WDIR/usr/bin/insmod
fi

# Add lvm symlinks if appropriate
# Also copy the lvm.conf file
if [ -x /usr/sbin/lvm ] ; then
    ln -s lvm $WDIR/usr/sbin/lvchange
    ln -s lvm $WDIR/usr/sbin/lvrename
    ln -s lvm $WDIR/usr/sbin/lvextend
    ln -s lvm $WDIR/usr/sbin/lvcreate
    ln -s lvm $WDIR/usr/sbin/lvdisplay
    ln -s lvm $WDIR/usr/sbin/lvscan

    ln -s lvm $WDIR/usr/sbin/pvchange
    ln -s lvm $WDIR/usr/sbin/pvck
    ln -s lvm $WDIR/usr/sbin/pvcreate
    ln -s lvm $WDIR/usr/sbin/pvdisplay
    ln -s lvm $WDIR/usr/sbin/pvscan

    ln -s lvm $WDIR/usr/sbin/vgchange
    ln -s lvm $WDIR/usr/sbin/vgcreate
    ln -s lvm $WDIR/usr/sbin/vgscan
    ln -s lvm $WDIR/usr/sbin/vgrename
    ln -s lvm $WDIR/usr/sbin/vgck
    # Conf file(s)
    cp -a /etc/lvm $WDIR/etc
fi

# Install libraries
sort $unsorted | uniq | while read library ; do
    # linux-vdso and linux-gate are pseudo libraries and do not correspond to a file
    # libsystemd-shared is in /lib/systemd, so it is not found by copy, and
    # it is copied below anyway
    if [[ "$library" == linux-vdso.so.1 ]] ||
       [[ "$library" == linux-gate.so.1 ]] ||
       [[ "$library" == libsystemd-shared* ]]; then
        continue
    fi

    copy $library lib
done

if [ -d /usr/lib/udev ]; then
    cp -a /usr/lib/udev $WDIR/usr/lib

```

```

fi
if [ -d /usr/lib/systemd ]; then
  cp -a /usr/lib/systemd $WDIR/usr/lib
fi
if [ -d /usr/lib/elogind ]; then
  cp -a /usr/lib/elogind $WDIR/usr/lib
fi

# Install the kernel modules if requested
if [ -n "$KERNEL_VERSION" ]; then
  find \
    /usr/lib/modules/$KERNEL_VERSION/kernel/{crypto,fs,lib} \
    /usr/lib/modules/$KERNEL_VERSION/kernel/drivers/{block,ata,nvme,md,firewire} \
    /usr/lib/modules/$KERNEL_VERSION/kernel/drivers/{scsi,message,pcmcia,virtio} \
    /usr/lib/modules/$KERNEL_VERSION/kernel/drivers/usb/{host,storage} \
    -type f 2>/dev/null | cpio --make-directories -p --quiet $WDIR

  cp /usr/lib/modules/$KERNEL_VERSION/modules.{builtin,order} \
      $WDIR/usr/lib/modules/$KERNEL_VERSION
  if [ -f /usr/lib/modules/$KERNEL_VERSION/modules.builtin.modinfo ]; then
    cp /usr/lib/modules/$KERNEL_VERSION/modules.builtin.modinfo \
        $WDIR/usr/lib/modules/$KERNEL_VERSION
  fi

  depmod -b $WDIR $KERNEL_VERSION
fi

( cd $WDIR ; find . | cpio -o -H newc --quiet | gzip -9 ) > $INITRAMFS_FILE

# Prepare early loading of microcode if available
if ls /usr/lib/firmware/intel-ucode/* >/dev/null 2>&1 ||
  ls /usr/lib/firmware/amd-ucode/* >/dev/null 2>&1; then

# first empty WDIR to reuse it
rm -r $WDIR/*

DSTDIR=$WDIR/kernel/x86/microcode
mkdir -p $DSTDIR

if [ -d /usr/lib/firmware/amd-ucode ]; then
  cat /usr/lib/firmware/amd-ucode/microcode_amd*.bin > $DSTDIR/AuthenticAMD.bin
fi

if [ -d /usr/lib/firmware/intel-ucode ]; then
  cat /usr/lib/firmware/intel-ucode/* > $DSTDIR/GenuineIntel.bin
fi

( cd $WDIR; find . | cpio -o -H newc --quiet ) > microcode.img
cat microcode.img $INITRAMFS_FILE > tmpfile
mv tmpfile $INITRAMFS_FILE
rm microcode.img
fi

# Remove the temporary directories and files
rm -rf $WDIR $unsorted
printf "done.\n"

EOF

chmod 0755 /usr/sbin/mkinitramfs

```

```

mkdir -p /usr/share/mkinitramfs &&
cat > /usr/share/mkinitramfs/init.in << "EOF"
#!/bin/sh

PATH=/usr/bin:/usr/sbin
export PATH

problem()
{
  printf "Encountered a problem!\n\nDropping you to a shell.\n\n"
  sh
}

no_device()
{

```

```

printf "The device %s, which is supposed to contain the\n" $1
printf "root file system, does not exist.\n"
printf "Please fix this problem and exit this shell.\n\n"
}

no_mount()
{
    printf "Could not mount device %s\n" $1
    printf "Sleeping forever. Please reboot and fix the kernel command line.\n\n"
    printf "Maybe the device is formatted with an unsupported file system?\n\n"
    printf "Or maybe filesystem type autodetection went wrong, in which case\n"
    printf "you should add the rootfstype=... parameter to the kernel command line.\n\n"
    printf "Available partitions:\n"
}

do_mount_root()
{
    mkdir /.root
    [ -n "$rootflags" ] && rootflags="$rootflags"
    rootflags="$rootflags$ro"

    case "$root" in
        /dev/*      ) device=$root ;;
        UUID=*     ) eval $root; device="/dev/disk/by-uuid/$UUID" ;;
        PARTUUID=* ) eval $root; device="/dev/disk/by-partuuid/$PARTUUID" ;;
        LABEL=*    ) eval $root; device="/dev/disk/by-label/$LABEL" ;;
        ""          ) echo "No root device specified." ; problem ;;
    esac

    while [ ! -b "$device" ] ; do
        no_device $device
        problem
    done

    if ! mount -n -t "$rootfstype" -o "$rootflags" "$device" /.root ; then
        no_mount $device
        cat /proc/partitions
        while true ; do sleep 10000 ; done
    else
        echo "Successfully mounted device $root"
    fi
}

do_try_resume()
{
    case "$resume" in
        UUID=* ) eval $resume; resume="/dev/disk/by-uuid/$UUID" ;;
        LABEL=* ) eval $resume; resume="/dev/disk/by-label/$LABEL" ;;
    esac

    if $noresume || ! [ -b "$resume" ]; then return; fi

    ls -lH "$resume" | ( read x x x x maj min x
                           echo -n ${maj%,}:$min > /sys/power/resume )
}

init=/sbin/init
root=
rootdelay=
rootfstype=auto
ro="ro"
rootflags=
device=
resume=
noresume=false

mount -n -t devtmpfs devtmpfs /dev
mount -n -t proc    proc      /proc
mount -n -t sysfs   sysfs    /sys
mount -n -t tmpfs   tmpfs    /run

read -r cmdline < /proc/cmdline

for param in $cmdline ; do
    case $param in
        init=*      ) init=${param#init=}           ;;

```

```

root=*      ) root=${param#root=}          ;;
rootdelay=* ) rootdelay=${param#rootdelay=} ;;
rootfstype=*) rootfstype=${param#rootfstype=} ;;
rootflags=* ) rootflags=${param#rootflags=} ;;
resume=*    ) resume=${param#resume=}       ;;
noresume    ) noresume=true                ;;
ro         ) ro="ro"                      ;;
rw         ) ro="rw"                      ;;

esac
done

# udevd location depends on version
if [ -x /sbin/udevd ]; then
  UDEVD=/sbin/udevd
elif [ -x /lib/udev/udevd ]; then
  UDEVD=/lib/udev/udevd
elif [ -x /lib/systemd/systemd-udevd ]; then
  UDEVD=/lib/systemd/systemd-udevd
else
  echo "Cannot find udevd nor systemd-udevd"
  problem
fi

${UDEVD} --daemon --resolve-names=never
udevadm trigger
udevadm settle

if [ -f /etc/mdadm.conf ] ; then mdadm -As           ; fi
if [ -x /sbin/vgchange ] ; then /sbin/vgchange -a y > /dev/null ; fi
if [ -n "$rootdelay" ] ; then sleep "$rootdelay"      ; fi

do_try_resume # This function will not return if resuming from disk
do_mount_root

killall -w ${UDEVD##*/}

exec switch_root /.root "$init" "$@"

EOF

```

Using an initramfs

Required Runtime Dependency

[cpio-2.15](#)

Other Runtime Dependencies

[LVM2-2.03.26](#) and/or [mdadm-4.3](#) must be installed before generating the initramfs, if the system partition uses them.

To build an initramfs, run the following as the `root` user:

```
mkinitsramfs [KERNEL VERSION]
```

The optional argument is the directory where the appropriate kernel modules are located. This must be a subdirectory of `/lib/modules`. If no modules are specified, then the initramfs is named `initrd.img-no-kmods`. If a kernel version is specified, the initrd is named `initrd.img-$KERNEL_VERSION` and is only appropriate for the specific kernel specified. The output file will be placed in the current directory.

If early loading of microcode is needed (see [the section called "Microcode updates for CPUs"](#)), you can install the appropriate blob or container in `/lib/firmware`. It will be automatically added to the initrd when running `mkinitsramfs`.

After generating the initrd, copy it to the `/boot` directory.

Now edit `/boot/grub/grub.cfg` and add a new menuentry. Below are several examples.

```

# Generic initramfs and root fs identified by UUID
menuentry "LFS Dev (LFS-7.0-Feb14) initrd, Linux 3.0.4"
{
  linux  /vmlinuz-3.0.4-1fs-20120214 root=UUID=54b934a9-302d-415e-ac11-4988408eb0a8 ro
  initrd /initrd.img-no-kmods
}

```

```

# Generic initramfs and root fs on LVM partition
menuentry "LFS Dev (LFS-7.0-Feb18) initrd lvm, Linux 3.0.4"
{
    linux /vmlinuz-3.0.4-lfs-20120218 root=/dev/mapper/myroot ro
    initrd /initrd.img-no-kmods
}

# Specific initramfs and root fs identified by LABEL
menuentry "LFS Dev (LFS-7.1-Feb20) initrd label, Linux 3.2.6"
{
    linux /vmlinuz-3.2.6-lfs71-120220 root=LABEL=lfs71 ro
    initrd /initrd.img-3.2.6-lfs71-120220
}

```

Finally, reboot the system and select the desired system.

btrfs-progs-6.10.1

Introduction to btrfs-progs

The btrfs-progs package contains administration and debugging tools for the B-tree file system (btrfs).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/kernel/people/kdave/btrfs-progs/btrfs-progs-v6.10.1.tar.xz>
- Download MD5 sum: a72e76a634ad70dd183153bb99034e4c
- Download size: 2.5 MB
- Estimated disk space required: 65 MB (transient files created during tests need up to 10 GB)
- Estimated build time: 0.3 SBU (with parallelism=4; add 4.0 SBU for tests, up to 14 SBU on slow disks)

Btrfs-progs Dependencies

Required

[LZO-2.10](#)

Optional

[LVM2-2.03.26](#) (`dmsetup` is used in tests), [sphinx-8.0.2](#) and [sphinx_rtd_theme-2.0.0](#) (required to build documentation), and [reiserfsprogs](#) (for tests).

Kernel Configuration

Enable the following option in the kernel configuration and recompile the kernel:

```

File systems --->
  <*/M> Btrfs filesystem support [BTRFS_FS]

```

In addition to the above and to the options required for [LVM2-2.03.26](#), the following options must be set for running tests. The options marked as empty should *not* be selected because they are for developers and make some tests fail.

```

File systems --->
  <*/M> Btrfs filesystem support [BTRFS_FS]
    [*]   Btrfs POSIX Access Control Lists [BTRFS_FS_POSIX_ACL]
    [ ]   Btrfs will run sanity tests upon loading [BTRFS_FS_RUN_SANITY_TESTS]
    [ ]   Btrfs debugging support [BTRFS_DEBUG]
    [ ]   Btrfs assert support [BTRFS_ASSERT]
    [ ]   Btrfs with the ref verify tool compiled in [BTRFS_FS_REF_VERIFY]

```

Installation of btrfs-progs

Install btrfs-progs by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --disable-documentation &&
make
```

Note

Some tests require grep built with perl regular expressions. To obtain this, rebuild grep with the LFS Chapter 8 instructions after installing [pcre2-10.44](#).

Before running tests, build a support program:

```
make fssum
```

To test the results, issue (as the `root` user):

```
pushd tests
./fsck-tests.sh
./mkfs-tests.sh
./cli-tests.sh
./convert-tests.sh
./misc-tests.sh
./fuzz-tests.sh
popd
```

Note

If the above mentioned kernel options are not enabled, some tests fail, and prevent all the remaining tests from running because the test disk image is not cleanly unmounted. Also make sure that the btrfs kernel module is loaded.

Install the package as the `root` user:

```
make install
```

If you have passed `--disable-documentation` to `configure` and you need the manual pages, install them by running, as the `root` user:

```
for i in 5 8; do
    install Documentation/*.$i /usr/share/man/man$i
done
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-documentation`: This switch disables rebuilding the manual pages, because it requires [sphinx-8.0.2](#).

Contents

Installed Programs: btrfs, btrfs-convert, btrfs-find-root, btrfs-image, btrfs-map-logical, btrfs-select-super, btrfsck (link to btrfs), btrfs-tune, fsck.btrfs, and mkfs.btrfs

Installed Libraries: libbtrfs.so and libbtrfsutil.so

Installed Directories: /usr/include/btrfs

Short Descriptions

<code>btrfs</code>	is the main interface into btrfs filesystem operations
<code>btrfs-convert</code>	converts from an ext2/3/4 or reiserfs filesystem to btrfs
<code>btrfs-find-root</code>	is a filter to find btrfs root
<code>btrfs-map-logical</code>	maps btrfs logical extent to physical extent
<code>btrfs-select-super</code>	overwrites the primary superblock with a backup copy
<code>btrfs-tune</code>	tunes various filesystem parameters

<code>fsck.btrfs</code>	does nothing, but is present for consistency with fstab
<code>mkfs.btrfs</code>	creates a btrfs file system

dosfstools-4.2

Introduction to dosfstools

The dosfstools package contains various utilities for use with the FAT family of file systems.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/dosfstools/dosfstools/releases/download/v4.2/dosfstools-4.2.tar.gz>
- Download MD5 sum: 49c8e457327dc61efab5b115a27b087a
- Download size: 314 KB
- Estimated disk space required: 4.5 MB (with tests)
- Estimated build time: less than 0.1 SBU

Kernel Configuration

Enable the following option in the kernel configuration and recompile the kernel if you need to mount a FAT-family filesystem:

```
File systems --->
  DOS/FAT/EXFAT/NT Filesystems --->
    < /*/M> MSDOS fs support           [MSDOS_FS]
    <*/M>   VFAT (Windows-95) fs support [VFAT_FS]
```

Note that `CONFIG_MSDOS_FS` deliberately does not support long file names. `CONFIG_VFAT_FS` should be used instead unless you really want to enforce the DOS-style "8.3" file names.

You can mount a FAT-family filesystem once the kernel supports it. If you don't need to create, check, or relabel a FAT-family system, you may skip this package.

Installation of dosfstools

Install dosfstools by running the following commands:

```
./configure --prefix=/usr          \
            --enable-compat-symlinks \
            --mandir=/usr/share/man \
            --docdir=/usr/share/doc/dosfstools-4.2 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-compat-symlinks`: This switch creates the dosfsck, dosfslabel, fsck.msdos, fsck.vfat, mkdosfs, mkfs.msdos, and mkfs.vfat symlinks required by some programs.

Contents

Installed Programs: fatlabel, fsck.fat, and mkfs.fat

Short Descriptions

<code>fatlabel</code>	sets or gets a MS-DOS filesystem label from a given device
<code>fsck.fat</code>	checks and repairs MS-DOS filesystems

`mkfs.fat` creates an MS-DOS filesystem under Linux

Fuse-3.16.2

Introduction to Fuse

FUSE (Filesystem in Userspace) is a simple interface for userspace programs to export a virtual filesystem to the Linux kernel. Fuse also aims to provide a secure method for non privileged users to create and mount their own filesystem implementations.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libfuse/libfuse/releases/download/fuse-3.16.2/fuse-3.16.2.tar.gz>
- Download MD5 sum: b00bf08b27ead4a9411578777e94a1cc
- Download size: 14 MB
- Estimated disk space required: 222 MB (with tests and documentation)
- Estimated build time: 0.2 SBU (add 0.3 SBU for tests)

Fuse Dependencies

Optional

[Doxygen-1.12.0](#) (to rebuild the API documentation), [pytest-8.3.2](#) (required for tests), and [looseversion](#) (for tests)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
File systems --->
  <*/M> FUSE (Filesystem in Userspace) support           [FUSE_FS]
```

Character devices in userspace should be enabled too for running the tests:

```
File systems --->
  <*/M> FUSE (Filesystem in Userspace) support           [FUSE_FS]
  <*/M>   Character device in Userspace support          [CUSE]
```

Installation of Fuse

Install Fuse by running the following commands:

```
sed -i '/^udev/, $ s/^/#/' util/meson.build &&

mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

The API documentation is included in the package, but if you have [Doxygen-1.12.0](#) installed and wish to rebuild it, issue:

```
pushd .. &&
doxygen doc/Doxyfile &&
popd
```

To test the results, issue the following commands (as the `root` user):

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install looseversion
python3 -m pytest
deactivate
```

The [pytest-8.3.2](#) Python module is required for the tests. One test named `test_cuse` will fail if the `CONFIG_CUSE` configuration item was not enabled when the kernel was built. One test, `test/util.py`, will output a warning due to the usage of an unknown mark in pytest.

Now, as the `root` user:

```
ninja install      &&
chmod u+s /usr/bin/fusermount3 &&

cd ..
cp -Rv doc/html -T /usr/share/doc/fuse-3.16.2 &&
install -v -m644 doc/{README.NFS,kernel.txt} \
/usr/share/doc/fuse-3.16.2
```

Command Explanations

`sed ... util/meson.build`: This command disables the installation of a boot script and udev rule that are not needed.

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Configuring fuse

Config Files

Some options regarding mount policy can be set in the file `/etc/fuse.conf`. To install the file run the following command as the `root` user:

```
cat > /etc/fuse.conf << "EOF"
# Set the maximum number of FUSE mounts allowed to non-root users.
# The default is 1000.
#
#mount_max = 1000

# Allow non-root users to specify the 'allow_other' or 'allow_root'
# mount options.
#
#user_allow_other
EOF
```

Additional information about the meaning of the configuration options are found in the man page.

Contents

Installed Programs: fusermount3 and mount.fuse3

Installed Libraries: libfuse3.so

Installed Directory: /usr/include/fuse3 and /usr/share/doc/fuse-3.16.2

Short Descriptions

<code>fusermount3</code>	is a uid root program to mount and unmount Fuse filesystems
<code>mount.fuse3</code>	is the command <code>mount</code> calls to mount a Fuse filesystem
<code>libfuse3.so</code>	contains the FUSE API functions

jfsutils-1.1.15

Introduction to jfsutils

The jfsutils package contains administration and debugging tools for the jfs file system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://jfs.sourceforge.net/project/pub/jfsutils-1.1.15.tar.gz>
- Download MD5 sum: 8809465cd48a202895bc2a12e1923b5d

- Download size: 532 KB
- Estimated disk space required: 8.9 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch to fix issues exposed by GCC 10 and later:
https://www.linuxfromscratch.org/patches/blfs/12.2/jfsutils-1.1.15-gcc10_fix-1.patch

Kernel Configuration

Enable the following option in the kernel configuration and recompile the kernel:

```
File systems --->
  <*/M> JFS filesystem support [JFS_FS]
```

Installation of jfsutils

First, fix some issues exposed by GCC 10 and later:

```
patch -Np1 -i .../jfsutils-1.1.15-gcc10_fix-1.patch
```

Install jfsutils by running the following commands:

```
sed -i "/unistd.h/a#include <sys/types.h>" fscklog/extract.c &&
sed -i "/ioctl.h/a#include <sys/sysmacros.h>" libfs/devices.c &&
./configure &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ...`: Fixes building with glibc 2.28.

Contents

Installed Programs: `fsck.jfs`, `jfs_debugfs`, `jfs_fsck`, `jfs_fscklog`, `jfs_logdump`, `jfs_mkfs`, `jfs_tune`, `mkfs.jfs`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>fsck.jfs</code>	is used to replay the JFS transaction log, check a JFS formatted device for errors, and fix any errors found
<code>jfs_fsck</code>	is a hard link to <code>fsck.jfs</code>
<code>mkfs.jfs</code>	constructs an JFS file system
<code>jfs_mkfs</code>	is a hard link to <code>mkfs.jfs</code>
<code>jfs_debugfs</code>	is a program which can be used to perform various low-level actions on a JFS formatted device
<code>jfs_fscklog</code>	extracts a JFS fsck service log into a file and/or formats and displays the extracted file
<code>jfs_logdump</code>	dumps the contents of the journal log from the specified JFS formatted device into output file <code>./jfslog.dmp</code>
<code>jfs_tune</code>	adjusts tunable file system parameters on JFS file systems

LVM2-2.03.26

Introduction to LVM2

The LVM2 package is a set of tools that manage logical partitions. It allows spanning of file systems across multiple physical disks and disk partitions and provides for dynamic growing or shrinking of logical partitions, mirroring and low storage footprint snapshots.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sourceware.org/ftp/lvm2/LVM2.2.03.26.tgz>
- Download MD5 sum: 276620e7071b4f0ba18b0c5cf2150c79
- Download size: 2.7 MB
- Estimated disk space required: 38 MB (add 20 MB for tests; transient files can grow up to around 300 MB in the /tmp directory during tests)
- Estimated build time: 0.3 SBU (add 9 to 48 SBU for tests, depending on disk speed and whether ram block device is enabled in the kernel)

LVM2 Dependencies

Required

[libaio-0.3.113](#)

Optional

[mdadm-4.3](#), [Valgrind-3.23.0](#), [Which-2.21](#), [xfsprogs-6.9.0](#) (all four may be used, but are not required, for tests), [reiserfsprogs](#), [thin-provisioning-tools](#), and [vdo](#)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel:

Note

There are several other Device Mapper options in the kernel beyond those listed below. In order to get reasonable results if running the regression tests, all must be enabled either internally or as a module. The tests will all time out if Magic SysRq key is not enabled.

```
Device Drivers --->
[*] Block devices --->
    <*/M> RAM block device support                                [BLK_DEV_RAM]
    [*] Multiple devices driver support (RAID and LVM) --->
        <*/M> Device mapper support                                     [MD]
        <*/M> Crypt target support                                    [BLK_DEV_DM]
        <*/M> Snapshot target                                       [DM_CRYPT]
        <*/M> Thin provisioning target                               [DM_SNAPSHOT]
        <*/M> Cache target (EXPERIMENTAL)                           [DM_THIN_PROVISIONING]
        <*/M> Mirror target                                         [DM_CACHE]
        <*/M> Zero target                                           [DM_MIRROR]
        <*/M> I/O delaying target                                   [DM_ZERO]
        <*/M> I/O delaying target                                   [DM_DELAY]

Kernel hacking --->
    Generic Kernel Debugging Instruments --->
        [*] Magic SysRq key                                         [MAGIC_SYSRQ]
```

Installation of LVM2

Install LVM2 by running the following commands:

```
PATH+=:/usr/sbin          \
./configure --prefix=/usr   \
            --enable-cmdlib \
            --enable-pkgconfig \
            --enable-udev_sync &&
make
```

The tests use udev for logical volume synchronization, so the LVM udev rules and some utilities need to be installed before running the tests. If you are installing LVM2 for the first time, and do not want to install the full package before running the

tests, the minimal set of utilities can be installed by running the following instructions as the `root` user:

```
make -C tools install_tools_dynamic &&
make -C udev install &&
make -C libdm install
```

To test the results, issue, as the `root` user:

```
LC_ALL=en_US.UTF-8 make check_local
```

Some tests may hang. In this case they can be skipped by adding `s=<testname>` to the `make` command. Other targets are available and can be listed with `make -C test help`. The test timings are very dependent on the speed of the disk(s), and on the number of enabled kernel options.

The tests do not implement the “expected fail” possibility, and a small number of test failures is expected by upstream. More failures may happen because some kernel options are missing. For example, the lack of the `dm-delay` device mapper target explains some failures. Some tests may fail if there is insufficient free space available in the partition with the `/tmp` directory. At least one test fails if 16 TB is not available. Some tests are flagged “warned” if [thin-provisioning-tools](#) are not installed. A workaround is to add the following flags to `configure`:

```
--with-thin-check= \
--with-thin-dump= \
--with-thin-repair= \
--with-thin-restore= \
--with-cache-check= \
--with-cache-dump= \
--with-cache-repair= \
--with-cache-restore= \
```

Some tests may hang. They can be removed if necessary, for example: `rm test/shell/lvconvert-raid-reshape.sh`. The tests generate a lot of kernel messages, which may clutter your terminal. You can disable them by issuing `dmesg -D` before running the tests (do not forget to issue `dmesg -E` when tests are done).

Note

The checks create device nodes in the `/tmp` directory. The tests will fail if `/tmp` is mounted with the `nodev` option.

Now, as the `root` user:

```
make install
make install_systemd_units
```

Command Explanations

`PATH+=:/usr/sbin`: The path must contain `/usr/sbin` for proper system tool detection by the `configure` script. This instruction ensures that PATH is properly set even if you build as an unprivileged user.

`--enable-cmdlib`: This switch enables building of the shared command library. It is required when building the event daemon.

`--enable-pkgconfig`: This switch enables installation of `pkg-config` support files.

`--enable-udev_sync`: This switch enables synchronisation with Udev processing.

`--enable-dmeventd`: This switch enables building of the Device Mapper event daemon.

`make install_systemd_units`: This is needed to install a unit that activates logical volumes at boot. It is not installed by default.

Configuring LVM2

Config File

```
/etc/lvm/lvm.conf
```

Configuration Information

The default configuration still references the obsolete `/var/lock` directory. This creates a deadlock at boot time. Change this (as the `root` user):

```
sed -e '/locking_dir =/{s/#//;s/var/run/}' \
-i /etc/lvm/lvm.conf
```

Contents

Installed Programs: `blkdeactivate`, `dmeventd` (optional), `dmsetup`, `fsadm`, `lvm`, `lvmdump`, and `lvm_import_vdo`. There are also numerous symbolic links to `lvm` that implement specific functionalities

Installed Libraries: `libdevmapper.so` and `liblvm2cmd.so`; optional: `libdevmapper-event.so`, `libdevmapper-event-lvm2.so`, `libdevmapper-event-lvm2mirror.so`, `libdevmapper-event-lvm2raid.so`, `libdevmapper-event-lvm2snapshot.so`, `libdevmapper-event-lvm2thin.so`, and `libdevmapper-event-lvm2vdo.so`

Installed Directories: `/etc/lvm` and `/usr/lib/device-mapper` (optional)

Short Descriptions

<code>blkdeactivate</code>	is a utility to deactivate block devices
<code>dmeventd</code>	(optional) is the Device Mapper event daemon
<code>dmsetup</code>	is a low level logical volume management tool
<code>fsadm</code>	is a utility used to resize or check filesystem on a device
<code>lvm</code>	provides the command-line tools for LVM2. Commands are implemented via symbolic links to this program to manage physical devices (<code>pv*</code>), volume groups (<code>vg*</code>) and logical volumes (<code>lv*</code>)
<code>lvmdump</code>	is a tool used to dump various information concerning LVM2
<code>vgimportclone</code>	is used to import a duplicated VG (e.g. hardware snapshot)
<code>libdevmapper.so</code>	contains the Device Mapper API functions

About Logical Volume Management (LVM)

LVM manages disk drives. It allows multiple drives and partitions to be combined into larger *volume groups*, assists in making backups through a *snapshot*, and allows for dynamic volume resizing. It can also provide mirroring similar to a RAID 1 array.

A complete discussion of LVM is beyond the scope of this introduction, but basic concepts are presented below.

To run any of the commands presented here, the [LVM2-2.03.26](#) package must be installed. All commands must be run as the `root` user.

Management of disks with lvm is accomplished using the following concepts:

physical volumes

These are physical disks or partitions such as `/dev/sda3` or `/dev/sdb`.

volume groups

These are named groups of physical volumes that can be manipulated by the administrator. The number of physical volumes that make up a volume group is arbitrary. Physical volumes can be dynamically added or removed from a volume group.

logical volumes

Volume groups may be subdivided into logical volumes. Each logical volume can then be individually formatted as if it were a regular Linux partition. Logical volumes may be dynamically resized by the administrator according to need.

To give a concrete example, suppose that you have two 2 TB disks. Also suppose a really large amount of space is required for a very large database, mounted on `/srv/mysql`. This is what the initial set of partitions would look like:

Partition	Use	Size	Partition Type
<code>/dev/sda1</code>	<code>/boot</code>	100MB	83 (Linux)
<code>/dev/sda2</code>	/	10GB	83 (Linux)
<code>/dev/sda3</code>	swap	2GB	82 (Swap)
<code>/dev/sda4</code>	LVM	remainder	8e (LVM)
<code>/dev/sdb1</code>	swap	2GB	82 (Swap)
<code>/dev/sdb2</code>	LVM	remainder	8e (LVM)

First initialize the physical volumes:

```
pvcreate /dev/sda4 /dev/sdb2
```

Note

A full disk can be used as part of a physical volume, but beware that the `pvcreate` command will destroy any partition information on that disk.

Next create a volume group named lfs-lvm:

```
vgcreate lfs-lvm /dev/sda4 /dev/sdb2
```

The status of the volume group can be checked by running the command `vgscan`. Now create the logical volumes. Since there is about 3900 GB available, leave about 900 GB free for expansion. Note that the logical volume named *mysql* is larger than any physical disk.

```
lvcreate --name mysql --size 2500G lfs-lvm
lvcreate --name home --size 500G lfs-lvm
```

Finally the logical volumes can be formatted and mounted. In this example, the jfs file system ([jfsutils-1.1.15](#)) is used for demonstration purposes.

```
mkfs -t ext4 /dev/lfs-lvm/home
mkfs -t jfs /dev/lfs-lvm/mysql
mount /dev/lfs-lvm/home /home
mkdir -p /srv/mysql
mount /dev/lfs-lvm/mysql /srv/mysql
```

It may be needed to activate those logical volumes, for them to appear in `/dev`. They can all be activated at the same time by issuing, as the `root` user:

```
vgchange -a y
```

A LVM logical volume can host a root filesystem, but requires the use of an initramfs (initial RAM file system). The initramfs proposed in [the section called "About initramfs"](#) allows to pass the lvm volume in the `root=` switch of the kernel command line.

If not using an initramfs, there is a race condition in systemd preventing mounting logical volumes through `/etc/fstab`. You must create a "mount" unit (see [systemd.mount\(5\)](#)) as in the following example, which mounts the `/home` directory automatically at boot:

```
cat > /etc/systemd/system/home.mount << EOF
[Unit]
Description=Mount the lvm volume /dev/lfs-lvm/home to /home

[Mount]
What=/dev/lfs-lvm/home
Where=/home
Type=ext4
Options=default

[Install]
WantedBy=multi-user.target
EOF
```

Note

The name of the unit must be the name of the mount point with the '/' character replaced by '-', omitting the leading one.

Next the unit must be enabled with:

```
systemctl enable home.mount
```

For more information about LVM, see the [LVM HOWTO](#) and the `lvm` man pages. A good in-depth [guide](#) is available from RedHat®, although it makes sometimes reference to proprietary tools.

The storage technology known as RAID (Redundant Array of Independent Disks) combines multiple physical disks into a logical unit. The drives can generally be combined to provide data redundancy or to extend the size of logical units beyond the capability of the physical disks or both. The technology also allows for providing hardware maintenance without powering down the system.

The types of RAID organization are described in the [RAID Wiki](#).

Note that while RAID provides protection against disk failures, it is not a substitute for backups. A file deleted is still deleted on all the disks of a RAID array. Modern backups are generally done via [rsync-3.3.0](#).

There are three major types of RAID implementation: Hardware RAID, BIOS-based RAID, and Software RAID.

Hardware RAID

Hardware based RAID provides capability through proprietary hardware and data layouts. The control and configuration is generally done via firmware in conjunction with executable programs made available by the device manufacturer. The capabilities are generally supplied via a PCI card, although there are some instances of RAID components integrated in to the motherboard. Hardware RAID may also be available in a stand-alone enclosure.

One advantage of hardware-based RAID is that the drives are offered to the operating system as a logical drive and no operating system dependent configuration is needed.

Disadvantages include difficulties in transferring drives from one system to another, updating firmware, or replacing failed RAID hardware.

BIOS-based RAID

Some computers offer a hardware-like RAID implementation in the system BIOS. Sometime this is referred to as 'fake' RAID as the capabilities are generally incorporated into firmware without any hardware acceleration.

The advantages and disadvantages of BIOS-based RAID are generally the same as hardware RAID with the additional disadvantage that there is no hardware acceleration.

In some cases, BIOS-based RAID firmware is enabled by default (e.g. some DELL systems). If software RAID is desired, this option must be explicitly disabled in the BIOS.

Software RAID

Software based RAID is the most flexible form of RAID. It is easy to install and update and provides full capability on all or part of any drives available to the system. In BLFS, the RAID software is found in [mdadm-4.3](#).

Configuring a RAID device is straightforward using mdadm. Generally devices are created in the `/dev` directory as `/dev/mdx` where x is an integer.

The first step in creating a RAID array is to use partitioning software such as [fdisk](#) or [parted-3.6](#) to define the partitions needed for the array. Usually, there will be one partition on each drive participating in the RAID array, but that is not strictly necessary. For this example, there will be four disk drives: `/dev/sda`, `/dev/sdb`, `/dev/sdc`, and `/dev/sdd`. They will be partitioned as follows:

Partition	Size	Type	Use
sdal:	100 MB	fd Linux raid auto	/boot (RAID 1) /dev/md0
sda2:	10 GB	fd Linux raid auto	/ (RAID 1) /dev/md1
sda3:	2 GB	83 Linux swap	swap
sda4	300 GB	fd Linux raid auto	/home (RAID 5) /dev/md2
sdb1:	100 MB	fd Linux raid auto	/boot (RAID 1) /dev/md0
sdb2:	10 GB	fd Linux raid auto	/ (RAID 1) /dev/md1
sdb3:	2 GB	83 Linux swap	swap
sdb4	300 GB	fd Linux raid auto	/home (RAID 5) /dev/md2
sdcl:	12 GB	fd Linux raid auto	/usr/src (RAID 0) /dev/md3
sdc2:	300 GB	fd Linux raid auto	/home (RAID 5) /dev/md2
sddl:	12 GB	fd Linux raid auto	/usr/src (RAID 0) /dev/md3
sdd2:	300 GB	fd Linux raid auto	/home (RAID 5) /dev/md2

In this arrangement, a separate boot partition is created as the first small RAID array and a root filesystem as the second RAID array, both mirrored. The third partition is a large (about 1TB) array for the `/home` directory. This provides an ability to stripe data across multiple devices, improving speed for both reading and writing large files. Finally, a fourth array is created that concatenates two partitions into a larger device.

Note

All mdadm commands must be run as the `root` user.

To create these RAID arrays the commands are:

```
/sbin/mdadm -Cv /dev/md0 --level=1 --raid-devices=2 /dev/sda1 /dev/sdb1
/sbin/mdadm -Cv /dev/md1 --level=1 --raid-devices=2 /dev/sda2 /dev/sdb2
/sbin/mdadm -Cv /dev/md3 --level=0 --raid-devices=2 /dev/sdc1 /dev/sdd1
/sbin/mdadm -Cv /dev/md2 --level=5 --raid-devices=4 \
    /dev/sda4 /dev/sdb4 /dev/sdc2 /dev/sdd2
```

The devices created can be examined by device. For example, to see the details of `/dev/md1`, use `/sbin/mdadm --detail /dev/md1`:

```
Version : 1.2
Creation Time : Tue Feb  7 17:08:45 2012
Raid Level : raid1
Array Size : 10484664 (10.00 GiB 10.74 GB)
Used Dev Size : 10484664 (10.00 GiB 10.74 GB)
Raid Devices : 2
Total Devices : 2
Persistence : Superblock is persistent

Update Time : Tue Feb  7 23:11:53 2012
State : clean
Active Devices : 2
Working Devices : 2
Failed Devices : 0
Spare Devices : 0

Name : core2-blfs:0  (local to host core2-blfs)
UUID : fcb944a4:9054aeb2:d987d8fe:a89121f8
Events : 17

Number  Major  Minor  RaidDevice State
      0      8      1          0  active sync  /dev/sda1
      1      8     17          1  active sync  /dev/sdb1
```

From this point, the partitions can be formatted with the filesystem of choice (e.g. ext3, ext4, [xfsprogs-6.9.0](#), etc). The formatted partitions can then be mounted. The `/etc/fstab` file can use the devices created for mounting at boot time and the linux command line in `/boot/grub/grub.cfg` can specify `root=/dev/md1`.

Note

The swap devices should be specified in the `/etc/fstab` file as normal. The kernel normally stripes swap data across multiple swap files and should not be made part of a RAID array.

For further options and management details of RAID devices, refer to `man mdadm`.

Additional details for monitoring RAID arrays and dealing with problems can be found at the [Linux RAID Wiki](#).

mdadm-4.3

Introduction to mdadm

The mdadm package contains administration tools for software RAID.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/utils/raid/mdadm/mdadm-4.3.tar.xz>
- Download MD5 sum: a42def84e31734a529111394f2289e0e
- Download size: 456 KB
- Estimated disk space required: 5.5 MB
- Estimated build time: 0.1 SBU

mdadm Dependencies

Optional

An [MTA](#)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel, if necessary. Only the RAID types desired are required.

```
Device Drivers --->
[*] Multiple devices driver support (RAID and LVM) ---> [MD]
  < */M>   RAID support [BLK_DEV_MD]
  [*]     Autodetect RAID arrays during kernel boot [MD_AUTODETECT]
# Only the RAID types desired are required:
< /*/M> RAID-0 (striping) mode [MD_RAID0]
< /*/M> RAID-1 (mirroring) mode [MD_RAID1]
< /*/M> RAID-10 (mirrored striping) mode [MD_RAID10]
< /*/M> RAID-4/RAID-5/RAID-6 mode [MD_RAID456]
```

Installation of mdadm

Build mdadm by running the following command:

```
make
```

This package does not come with a working test suite.

Now, as the `root` user:

```
make BINDIR=/usr/sbin install
```

Command Explanations

`make everything`: This optional target creates extra programs, particularly a statically-linked version of `mdadm`. This needs to be manually installed.

Contents

Installed Programs: `mdadm` and `mdmon`

Installed Libraries: None

Installed Directory: None

Short Descriptions

<code>mdadm</code>	manages MD devices aka Linux Software RAID
<code>mdmon</code>	monitors MD external metadata arrays

ntfs-3g-2022.10.3

Introduction to Ntfs-3g

Note

A new read-write driver for NTFS, called NTFS3, has been added into the Linux kernel since the 5.15 release. The performance of NTFS3 is much better than ntfs-3g. To enable NTFS3, enable the following options in the kernel configuration and recompile the kernel if necessary:

```
File systems --->
DOS/FAT/EXFAT/NT Filesystems --->
  <*/M> NTFS Read-Write file system support [NTFS3_FS]
```

To ensure the `mount` command uses NTFS3 for ntfs partitions, create a wrapper script:

```
cat > /usr/sbin/mount.ntfs << "EOF" &&
#!/bin/sh
exec mount -t ntfs3 "$@"
```

```
EOF  
chmod -v 755 /usr/sbin/mount.ntfs
```

With the kernel support available, ntfs-3g is only needed if you need the utilities from it (for example, to create NTFS filesystems).

The Ntfs-3g package contains a stable, read-write open source driver for NTFS partitions. NTFS partitions are used by most Microsoft operating systems. Ntfs-3g allows you to mount NTFS partitions in read-write mode from your Linux system. It uses the FUSE kernel module to be able to implement NTFS support in userspace. The package also contains various utilities useful for manipulating NTFS partitions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://tuxera.com/opensource/ntfs-3g_ntfsprogs-2022.10.3.tgz
- Download MD5 sum: a038af61be7584b79f8922ff11244090
- Download size: 1.3 MB
- Estimated disk space required: 22 MB
- Estimated build time: 0.2 SBU

Ntfs-3g Dependencies

Optional

[fuse 2.x](#) (this disables user mounts)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
File systems --->  
<*/M> FUSE (Filesystem in Userspace) support [FUSE_FS]
```

Note that it is only needed for mounting NTFS partitions with ntfs-3g. If you will use the in-kernel NTFS3 driver for mounting NTFS partitions (as the BLFS editors recommend) instead, you can skip this configuration item.

Installation of Ntfs-3g

Install Ntfs-3g by running the following commands:

```
./configure --prefix=/usr      \  
          --disable-static    \  
          --with-fuse=internal \  
          --docdir=/usr/share/doc/ntfs-3g-2022.10.3 &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
```

It's recommended to use the in-kernel NTFS3 driver for mounting NTFS filesystems, instead of ntfs-3g (see the note at the start of this page). However, if you want to use ntfs-3g to mount the NTFS filesystems anyway, create a symlink for the `mount` command:

```
ln -sv ../bin/ntfs-3g /usr/sbin/mount.ntfs &&  
ln -sv ntfs-3g.8 /usr/share/man/man8/mount.ntfs.8
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-fuse=internal`: This switch dynamically forces ntfs-3g to use an internal copy of the fuse-2.x library. This is required if you wish to allow users to mount NTFS partitions.

`--disable-ntfsprogs`: Disables installation of various utilities used to manipulate NTFS partitions.

`chmod -v 4755 /usr/bin/ntfs-3g`: Making mount.ntfs setuid root allows non root users to mount NTFS partitions.

Using Ntfs-3g

To mount a Windows partition at boot time, put a line like this in /etc/fstab:

```
/dev/sdal /mnt/windows auto defaults 0 0
```

To allow users to mount a usb stick with an NTFS filesystem on it, put a line similar to this (change sdc1 to whatever a usb stick would be on your system) in /etc/fstab:

```
/dev/sdc1 /mnt/usb auto user,noauto,umask=0,utf8 0 0
```

In order for a user to be able to mount the usb stick, they will need to be able to write to `/mnt/usb`, so as the `root` user:

```
chmod -v 777 /mnt/usb
```

Contents

Installed Programs: lowntfs-3g, mkfs.ntfs, mknntfs, mount.lowntfs-3g, mount.ntfs, mount.ntfs-3g, ntfs-3g, ntfs-3g.probe, ntfcscat, ntfsclone, ntfscluster, ntfsncmp, ntfsycop, ntfsfix, ntfsinfo, ntfslabel, ntfsls, ntfsresize and ntfsundelete

Installed Library: libntfs-3g.so

Installed Directories: /usr/include/ntfs-3g and /usr/share/doc/ntfs-3g

Short Descriptions

lowntfs-3g	is similar to ntfs-3g but uses the Fuse low-level interface
mkfs.ntfs	is a symlink to mkntfs
mkntfs	creates an NTFS file system
mount.lowntfs-3g	is a symlink to lowntfs-3g
mount.ntfs	mounts an NTFS filesystem
mount.ntfs-3g	is a symbolic link to ntfs-3g
ntfs-3g	is an NTFS driver, which can create, remove, rename, move files, directories, hard links, and streams. It can also read and write files, including streams, sparse files and transparently compressed files. It can also handle special files like symbolic links, devices, and FIFOs; moreover it provides standard management of file ownership and permissions, including POSIX ACLs
ntfs-3g.probe	tests if an NTFS volume is mountable read only or read-write, and exits with a status value accordingly. The volume can be a block device or image file
ntfscluster	identifies files in a specified region of an NTFS volume
ntfscp	copies a file to an NTFS volume
ntfsfix	fixes common errors and forces Windows to check an NTFS partition
ntfsls	lists directory contents on an NTFS filesystem
ntfscat	prints NTFS files and streams on the standard output
ntfsclone	clones an NTFS filesystem
ntfscmp	compares two NTFS filesystems and shows the differences
ntfsinfo	dumps a file's attributes
ntfslabel	displays or changes the label on an ntfs file system
ntfsresize	resizes an NTFS filesystem without data loss
ntfsundelete	recovers a deleted file from an NTFS volume
libntfs-3g.so	contains the Ntfs-3g API functions

gptfdisk-1.0.10

Introduction to qptfdisk

The `gptfdisk` package is a set of programs for creation and maintenance of GUID Partition Table (GPT) disk drives. A GPT partitioned disk is required for drives greater than 2 TB and is a modern replacement for legacy PC-BIOS partitioned disk drives that use a Master Boot Record (MBR). The main program, `gdisk`, has an interface similar to the classic `fdisk` program.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/gptfdisk/gptfdisk-1.0.10.tar.gz>
- Download MD5 sum: 1970269eb7a97560e238611524b7797a
- Download size: 216 KB
- Estimated disk space required: 2.4 MB
- Estimated build time: less than 0.1 SBU (add 0.2 SBU for tests)

Additional Downloads

- Recommended patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/gptfdisk-1.0.10-convenience-1.patch>

gptfdisk Dependencies

Required

[popt-1.19](#)

Optional

[ICU-75.1](#)

Installation of gptfdisk

The gptfdisk package comes with a rudimentary `Makefile`. First we update it to provide a simple build and install interface and fix the location of a header file and fix some minor location issues. Install gptfdisk by running the following commands:

```
patch -Np1 -i ../gptfdisk-1.0.10-convenience-1.patch &&
sed -i 's|ncursesw||' gptcurses.cc &&
sed -i 's|sbin|usr/sbin|' Makefile &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`patch -Np1 ...`: This patch modifies the `Makefile` file so that it provides an "install" target.

Contents

Installed Programs: `cgdisk`, `gdisk`, `fixparts`, and `sgdisk`

Short Descriptions

<code>cgdisk</code>	is an ncurses-based tool for manipulating GPT partitions
<code>gdisk</code>	is an interactive text-mode tool for manipulating GPT partitions
<code>fixparts</code>	repairs mis-formatted MBR based disk partitions
<code>sgdisk</code>	is a partition manipulation program for GPT partitions similar to <code>sfdisk</code>

parted-3.6

Introduction to parted

The Parted package is a disk partitioning and partition resizing tool.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/parted/parted-3.6.tar.xz>
- Download MD5 sum: 93d2d8f22baebc5eb65b85da05a79e4e
- Download size: 1.8 MB
- Estimated disk space required: 29 MB (additional 3 MB for the tests and additional 2 MB for optional PDF and Postscript documentation)
- Estimated build time: 0.3 SBU (additional 3.6 SBU for the tests)

Parted Dependencies

Recommended

[LVM2-2.03.26](#) (device-mapper, required if building udisks)

Optional

[dosfstools-4.2](#), [texlive-20240312](#) (or [install-tl-unx](#)), and [Digest::CRC](#) (for tests)

Optional Kernel Configuration for Tests

About 20 % more tests are run if the following kernel module is built:

```
Device Drivers --->
  SCSI device support --->
    [*] SCSI low-level drivers --->           [SCSI_LOWLEVEL]
    <M> SCSI debugging host and device simulator [SCSI_DEBUG]
```

Installation of parted

Install Parted by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make &&

make -C doc html &&
makeinfo --html -o doc/html doc/parted.texi &&
makeinfo --plaintext -o doc/parted.txt doc/parted.texi
```

If you have [texlive-20240312](#) installed and wish to create PDF and Postscript documentation issue the following commands:

```
cp build-aux/texinfo.tex doc &&
texi2pdf -o doc/parted.pdf doc/parted.texi &&
texi2dvi -o doc/parted.dvi doc/parted.texi &&
dvips -o doc/parted.ps doc/parted.dvi
```

To test the results, issue, as the `root` user:

```
make check
```

Note

Many tests are skipped if not run as the `root` user.

Now, as the `root` user:

```
make install &&
install -v -m755 -d /usr/share/doc/parted-3.6/html &&
install -v -m644 doc/html/* \
  /usr/share/doc/parted-3.6/html &&
install -v -m644 doc/{FAT,API,parted.{txt,html}} \
  /usr/share/doc/parted-3.6
```

Install the optional PDF and Postscript documentation by issuing the following command as the `root` user:

```
install -v -m644 doc/FAT doc/API doc/parted.{pdf,ps,dvi} \
/usr/share/doc/parted-3.6
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

--disable-device-mapper: This option disables device mapper support. Add this parameter if you have not installed LVM2.

Contents

Installed Programs: parted and partprobe

Installed Libraries: libparted.so and libparted-fs-resize.so

Installed Directories: /usr/include/parted and /usr/share/doc/parted-3.6

Short Descriptions

parted	is a partition manipulation program
partprobe	informs the OS of partition table changes
libparted.so	contains the Parted API functions

smartmontools-7.4

Introduction to smartmontools

The smartmontools package contains utility programs (smartctl, smartd) to control/monitor storage systems using the Self-Monitoring, Analysis and Reporting Technology System (S.M.A.R.T.) built into most modern ATA and SCSI disks.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/smartmontools/smartmontools-7.4.tar.gz>
- Download MD5 sum: 178d31a6ff5256c093227ab45a3f52aa
- Download size: 1 MB
- Estimated disk space required: 27 MB
- Estimated build time: 0.2 SBU

smartmontools Dependencies

Optional (runtime)

[curl-8.9.1](#) or [lynx-2.9.2](#) or [wget-1.24.5](#) (download tools), and [GnuPG-2.4.5](#) (encrypted hard disks)

Installation of smartmontools

Install smartmontools by running the following commands:

```
./configure --prefix=/usr           \
            --sysconfdir=/etc        \
            --docdir=/usr/share/doc/smartmontools-7.4 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Configuring smartmontools

Config File

/etc/smartd.conf

Configuration Information

See the embedded comments in /etc/smartd.conf for detailed instructions on customizing the smartd daemon.

Systemd Unit

If you want the smartd daemon to start automatically when the system is booted, enable the systemd unit provided by the package by executing the following command as the `root` user:

```
systemctl enable smartd
```

Contents

Installed Programs: smartctl, smartd, and update-smart-drive

Installed Libraries: None

Installed Directories: /usr/share/smartmontools, /usr/share/doc/smartmontools-7.4, and /etc/smartd_warning.d

Short Descriptions

<code>smartctl</code>	is the control and monitor utility for SMART Disks
<code>smartd</code>	is the SMART disk monitoring daemon
<code>update-smart-drive</code>	is the update tool for the smartmontools drive database

sshfs-3.7.3

Introduction to Sshfs

The Sshfs package contains a filesystem client based on the SSH File Transfer Protocol. This is useful for mounting a remote computer that you have ssh access to as a local filesystem. This allows you to drag and drop files or run shell commands on the remote files as if they were on your local computer.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libfuse/sshfs/releases/download/sshfs-3.7.3/sshfs-3.7.3.tar.xz>
- Download MD5 sum: f704f0d1800bdb5214030a1603e8c6d6
- Download size: 56 KB
- Estimated disk space required: 0.9 MB
- Estimated build time: less than 0.1 SBU

Sshfs Dependencies

Required

[Fuse-3.16.2](#), [GLib-2.80.4](#), and [OpenSSH-9.8p1](#).

Optional

[docutils-0.21.2](#) (required to build the man page)

Installation of Sshfs

Install Sshfs by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Using Sshfs

To mount an ssh server you need to be able to log into the server. For example, to mount your remote home folder to the local `~/examplepath` (the directory must exist and you must have permissions to write to it):

```
sshfs example.com:/home/userid ~/examplepath
```

When you've finished work and want to unmount it again:

```
fusermount3 -u ~/example
```

You can also mount an sshfs filesystem at boot by adding an entry similar to the following in the `/etc/fstab` file:

```
userid@example.com:/path /media/path fuse.sshfs _netdev,IdentityFile=/home/userid/.ssh/id_rsa 0 0
```

See `man 1 sshfs` and `man 8 mount.fuse3` for all available mount options.

Contents

Installed Program: `sshfs`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`sshfs` mounts an `ssh` server as a local file system

xfsprogs-6.9.0

Introduction to xfsprogs

The `xfsprogs` package contains administration and debugging tools for the XFS file system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/utils/fs/xfs/xfsprogs/xfsprogs-6.9.0.tar.xz>
- Download MD5 sum: 8744b22c73764320bcd577d98dbc4f2
- Download size: 1.3 MB
- Estimated disk space required: 57 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

xfsprogs Dependencies

Required

`inith-58` and `liburcu-0.14.0`

Optional

`ICU-75.1` (for unicode name scanning in `xfs_scrub`)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel:

```
File systems --->
  <*/M> XFS filesystem support           [XFS_FS]
```

Installation of xfsprogs

Install xfsprogs by running the following commands:

```
make DEBUG=--DNDEBUG \
      INSTALL_USER=root \
      INSTALL_GROUP=root
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PKG_DOC_DIR=/usr/share/doc/xfsprogs-6.9.0 install    &&
make PKG_DOC_DIR=/usr/share/doc/xfsprogs-6.9.0 install-dev &&

rm -rfv /usr/lib/libhandle.{a,la}
```

Command Explanations

`DEBUG=--DNDEBUG`: Turns off debugging symbols.

`INSTALL_USER=root` `INSTALL_GROUP=root`: This sets the owner and group of the installed files.

`OPTIMIZER="..."`: Adding this parameter to the end of the `make` command overrides the default optimization settings.

Contents

Installed Programs: `fsck.xfs`, `mkfs.xfs`, `xfs_admin`, `xfs_bmap`, `xfs_copy`, `xfs_db`, `xfs_estimate`, `xfs_freeze`, `xfs_fsr`, `xfs_growfs`, `xfs_info`, `xfs_io`, `xfs_logprint`, `xfs_mdrestore`, `xfs_metadump`, `xfs_mkfile`, `xfs_ncheck`, `xfs_quota`, `xfs_repair`, `xfs_rtcp`, `xfs_scrub`, `xfs_scrub_all`, and `xfs_spaceman`

Installed Libraries: `libhandle.so`

Installed Directories: `/usr/include/xfs`, `/usr/lib/xfsprogs`, `/usr/share/xfsprogs`, and `/usr/share/doc/xfsprogs-6.9.0`

Short Descriptions

<code>fsck.xfs</code>	simply exits with a zero status, since XFS partitions are checked at mount time
<code>mkfs.xfs</code>	constructs an XFS file system
<code>xfs_admin</code>	changes the parameters of an XFS file system
<code>xfs_bmap</code>	prints block mapping for an XFS file
<code>xfs_copy</code>	copies the contents of an XFS file system to one or more targets in parallel
<code>xfs_estimate</code>	for each directory argument, estimates the space that directory would take if it were copied to an XFS filesystem (does not cross mount points)
<code>xfs_db</code>	is used to debug an XFS file system
<code>xfs_freeze</code>	suspends access to an XFS file system
<code>xfs_fsr</code>	applicable only to XFS filesystems, improves the organization of mounted filesystems, the reorganization algorithm operates on one file at a time, compacting or otherwise improving the layout of the file extents (contiguous blocks of file data)
<code>xfs_growfs</code>	expands an XFS file system
<code>xfs_info</code>	is equivalent to invoking <code>xfs_growfs</code> , but specifying that no change to the file system is to be made
<code>xfs_io</code>	is a debugging tool like <code>xfs_db</code> , but is aimed at examining the regular file I/O path rather than the raw XFS volume itself
<code>xfs_logprint</code>	prints the log of an XFS file system
<code>xfs_mdrestore</code>	restores an XFS metadump image to a filesystem image
<code>xfs_metadump</code>	copies XFS filesystem metadata to a file
<code>xfs_mkfile</code>	creates an XFS file, padded with zeroes by default
<code>xfs_ncheck</code>	generates pathnames from inode numbers for an XFS file system
<code>xfs_quota</code>	is a utility for reporting and editing various aspects of filesystem quotas
<code>xfs_repair</code>	repairs corrupt or damaged XFS file systems
<code>xfs_rtcp</code>	copies a file to the real-time partition on an XFS file system
<code>xfs_scrub</code>	checks and repairs the contents of a mounted XFS file system
<code>xfs_scrub_all</code>	scrubs all mounted XFS file systems
<code>xfs_spaceman</code>	reports and controls free space usage in an XFS file system

Packages for UEFI Boot

efivar-39

Introduction to efivar

The efivar package provides tools and libraries to manipulate EFI variables.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/rhboot/efivar/archive/39/efivar-39.tar.gz>
- Download MD5 sum: a8fc3e79336cd6e738ab44f9bc96a5aa
- Download size: 456 KB
- Estimated disk space required: 21 MB
- Estimated build time: less than 0.1 SBU

efivar Dependencies

Recommended

[mandoc-1.14.6](#)

Installation of efivar

Note

This package cannot function properly on a 32-bit system with a 64-bit UEFI implementation. Don't install this package (or efibootmgr) on 32-bit system unless you are absolutely sure you have a 32-bit UEFI implementation, which is very rare in practice.

Build efivar with the following commands:

```
make
```

The test suite of this package is dangerous. Running it may trigger firmware bugs and make your system unusable without using some special hardware to reprogram the firmware.

Now, as the `root` user:

```
make install LIBDIR=/usr/lib
```

Command Explanations

`LIBDIR=/usr/lib`: This option overrides the default library directory of the package (`/usr/lib64`, which is not used by LFS).

`ENABLE_DOCS=0`: Disable the generation of man pages. Append this option after the `make` and `make install` commands if you don't need the man pages to allow building this package without [mandoc-1.14.6](#) installed.

Contents

Installed Programs: efisecdb and efivar

Installed Libraries: libefiboot.so, libefisec.so, and libefivar.so

Installed Directories: /usr/include/efivar

Short Descriptions

<code>efisecdb</code>	is an utility for managing UEFI signature lists
<code>efivar</code>	is a tool to manipulate UEFI variables
<code>libefiboot.so</code>	is a library used by <code>efibootmgr</code>
<code>libefisec.so</code>	is a library for managing UEFI signature lists
<code>libefivar.so</code>	is a library for the manipulation of EFI variables

efibootmgr-18

Introduction to efibootmgr

The efibootmgr package provides tools and libraries to manipulate EFI variables.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/rhboot/efibootmgr/archive/18/efibootmgr-18.tar.gz>
- Download MD5 sum: e170147da25e1d5f72721ffc46fe4e06
- Download size: 48 KB
- Estimated disk space required: 1.1 MB
- Estimated build time: less than 0.1 SBU

efibootmgr Dependencies

Required

[efivar-39](#) and [popt-1.19](#)

Installation of efibootmgr

Build efibootmgr with the following commands:

```
make EFIDIR=LFS EFI_LOADER=grubx64.efi
```

This package does not have a test suite.

Now, as the `root` user:

```
make install EFIDIR=LFS
```

Command Explanations

`EFIDIR=LFS`: This option specifies the distro's subdirectory name under `/boot/efi/EFI`. The building system of this package needs it to be set explicitly.

`EFI_LOADER=grubx64.efi`: This option specifies the name of the default EFI boot loader. It is set to match the EFI boot loader provided by GRUB .

Contents

Installed Programs: efibootdump and efibootmgr

Short Descriptions

<code>efibootdump</code>	is a tool to display individual UEFI boot options, from a file or an UEFI variable
<code>efibootmgr</code>	is a tool to manipulate the UEFI Boot Manager

GRUB-2.12 for EFI

Introduction to GRUB

The GRUB package provides GRand Unified Bootloader. In this page it will be built with UEFI support, which is not enabled for GRUB built in LFS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/grub/grub-2.12.tar.xz>
- Download MD5 sum: 60c564b1bdc39d8e43b3aab4bc0fb140
- Download size: 6.4 MB
- Estimated disk space required: 183 MB
- Estimated build time: 0.4 SBU (on 64-bit LFS, using parallelism=4)

Additional Downloads

Unicode font data used to display GRUB menu

- Download (HTTP): <https://unifoundry.com/pub/unifont/unifont-15.1.05/font-builds/unifont-15.1.05.pcf.gz>
- Download MD5 sum: da47e9c7a2cec3b68a0fad5d2a341dcc
- Download size: 1.4 MB

GCC (only needed if building on 32-bit LFS)

- Refer to [GCC-14.2.0](#) page for download info.

GRUB Dependencies

Recommended

[efibootmgr-18](#) (runtime) and [FreeType-2.13.3](#)

Optional

[LVM2-2.03.26](#) and [Fuse-3.16.2](#)

Installation of GRUB

First, install font data as the `root` user:

```
mkdir -pv /usr/share/fonts/unifont &&
gunzip -c ../unifont-15.1.05.pcf.gz > /usr/share/fonts/unifont/unifont.pcf
```

Warning

Unset any environment variables which may affect the build:

```
unset {C,CPP,CXX,LD}FLAGS
```

Don't try "tuning" this package with custom compilation flags: this package is a bootloader, with low-level operations in the source code which is likely to be broken by some aggressive optimizations.

Add a file missing from the release tarball:

```
echo depends bli part_gpt > grub-core/extra_deps.lst
```

If you are running a 32-bit LFS, prepare a 64-bit compiler:

```
case $(uname -m) in i?86 )
    tar xf ../gcc-14.2.0.tar.xz
    mkdir gcc-14.2.0/build
    pushd gcc-14.2.0/build
    ./configure --prefix=$PWD/../../x86_64-gcc \
                --target=x86_64-linux-gnu \
                --with-system-zlib \
                --enable-languages=c,c++ \
                --with-ld=/usr/bin/ld
```

```

        make all-gcc
        make install-gcc
    popd
    export TARGET_CC=$PWD/x86_64-gcc/bin/x86_64-linux-gnu-gcc
esac

```

Build GRUB with the following commands:

```

./configure --prefix=/usr      \
--sysconfdir=/etc      \
--disable-efemu      \
--enable-grub-mkfont \
--with-platform=efi   \
--target=x86_64      \
--disable-werror      &&
unset TARGET_CC &&
make

```

This package does not have a test suite providing meaningful results.

Now, if you've skip the LFS GRUB package, as the `root` user:

```

make install &&
mv -v /etc/bash_completion.d/grub /usr/share/bash-completion/completions

```

If you've not skip LFS GRUB package, as the `root` user, only install the components not installed from the LFS GRUB package instead:

```

make DESTDIR=$PWD/dest install
cp -av dest/usr/lib/grub/x86_64-efi -T /usr/lib/grub/x86_64-efi
cp -av dest/usr/share/grub/*.pf2,h /usr/share/grub
cp -av dest/usr/bin/grub-mkfont /usr/bin

```

If the optional dependencies are installed, also install the `grub-mount` program:

```

cp -av dest/usr/bin/grub-mount /usr/bin

```

Command Explanations

`--enable-grub-mkfont`: Build the tool named `grub-mkfont` to generate the font file for the boot loader from the font data we've installed.

Warning

If the recommended dependency [FreeType-2.13.3](#) is not installed, it is possible to omit this option and build GRUB. However, if `grub-mkfont` is not built, or the unicode font data is not available at the time GRUB is built, GRUB won't install any font for the boot loader. The GRUB boot menu will be displayed using a coarse font or in a smaller region on the screen.

`--with-platform=efi`: Ensures building GRUB with EFI enabled.

`--target=x86_64`: Ensures building GRUB for x86_64 even if building on a 32-bit LFS system. Most EFI firmware on x86_64 does not support 32-bit bootloaders.

`--target=i386`: A few 32-bit x86 platforms have EFI support. And, some x86_64 platforms have a 32-bit EFI implementation, but they are very old and rare. Use this instead of `--target=x86_64` if you are **absolutely sure** that LFS is running on such a system.

Configuring GRUB

Using GRUB to make the LFS system bootable on UEFI platform will be discussed in [Using GRUB to Set Up the Boot Process with UEFI](#).

Contents

A list of the installed files, along with their short descriptions can be found at [..../.../.../lfs/view/12.2-systemd/chapter08/grub.html#contents-grub](#).

Listed below are the newly installed programs along with short descriptions.

Installed Programs: grub-mkfont and grub-mount (optional)

Installed Directories: /usr/lib/grub/x86_64-efi

Short Descriptions

grub-mkfont	converts common font file formats into PF2
grub-mount	is a debug tool for filesystem driver

Using GRUB to Set Up the Boot Process with UEFI

Turn Off Secure Boot

BLFS does not have the essential packages to support Secure Boot. To set up the boot process with GRUB and UEFI in BLFS, Secure Boot must be turned off from the configuration interface of the firmware. Read the documentation provided by the manufacturer of your system to find out how.

Kernel Configuration for UEFI support

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
Processor type and features --->
  [*] EFI runtime service support [EFI]
  [*]   EFI stub support [EFI_STUB]

--> Enable the block layer ---> [BLOCK]
  Partition Types --->
    [ /*] Advanced partition selection [PARTITION_ADVANCED]
    [*]   EFI GUID Partition support [EFI_PARTITION]

Device Drivers --->
  Firmware Drivers --->
    [*] Mark VGA/VBE/EFI FB as generic system framebuffer [SYSFB_SIMPLEFB]
  Graphics support --->
    <*> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
      ...
      [*] Enable legacy fbdev support for your modesetting driver ... [DRM_FBDEV_EMULATION]
      <>> Simple framebuffer driver [DRM_SIMPLEDRM]
    Console display driver support --->
      [*] Framebuffer Console support [FRAMEBUFFER_CONSOLE]

File systems --->
  DOS/FAT/EXFAT/NT Filesystems --->
    <*/M> VFAT (Windows-95) fs support [VFAT_FS]
  Pseudo filesystems --->
    <*/M> EFI Variable filesystem [EFIVAR_FS]
--> Native language support --->
  <*/M> Codepage 437 (United States, Canada) [NLS_CODEPAGE_437]
  <*/M> NLS ISO 8859-1 (Latin 1; Western European Languages) [NLS_ISO8859_1]
```

The meaning of the configuration options:

`CONFIG_PARTITION_ADVANCED`

If it's not enabled, `CONFIG_EFI_PARTITION` will be enabled automatically. But when it's enabled, you must set `CONFIG_EFI_PARTITION` to enabled as well.

`CONFIG_SYSFB_SIMPLEFB, CONFIG_DRM, CONFIG_DRM_FBDEV_EMULATION, CONFIG_DRM_SIMPLEDRM, CONFIG_FB, and CONFIG_FRAMEBUFFER_CONSOLE`

The combination of these options provides the Linux console support on top of the UEFI framebuffer. To allow the kernel to print debug messages at an early boot stage, they shouldn't be built as kernel modules unless an initramfs will be used.

Create an Emergency Boot Disk

Ensure that an emergency boot disk is ready to "rescue" the system in case the system becomes un-bootable. To make an emergency boot disk with GRUB for an EFI based system, find a spare USB flash drive and create a `vfat` file system on it.

Install [dosfstools-4.2](#) first, then as the `root` user:

Warning

The following command will erase all directories and files in the partition. Make sure your USB flash drive contains no data which will be needed, and change `sdx1` to the device node corresponding to the first partition of the USB flash drive. Be careful not to overwrite your hard drive with a typo!

```
mkfs.vfat /dev/sdx1
```

Still as the `root` user, use the `fdisk` utility to set the first partition of the USB flash drive to be an "EFI system" partition (change `sdx` to the device node corresponding to your USB flash drive):

```
fdisk /dev/sdx

Welcome to fdisk (util-linux 2.39.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): t
Partition number (1-9, default 9): 1
Partition type or alias (type L to list all): uefi
Changed type of partition 'Linux filesystem' to 'EFI System'.

Command (m for help): w
The partition table has been altered.
Syncing disks.
```

Still as the `root` user, create a mount point for the EFI partition on the USB flash drive and mount it:

```
mount --mkdir -v -t vfat /dev/sdx1 -o codepage=437,iocharset=iso8859-1 \
/mnt/rescue
```

Install GRUB for EFI on the partition:

```
grub-install --target=x86_64-efi --removable \
--efi-directory=/mnt/rescue --boot-directory=/mnt/rescue
```

Unmount the partition:

```
umount /mnt/rescue
```

Now the USB flash drive can be used as an emergency boot disk on any x86-64 UEFI platform. To learn how to select this flash drive as the boot device, read the manual of your motherboard or laptop. It will boot the system and show the GRUB shell. Then you can type commands to boot your operating system from the hard drive.

Find or Create the EFI System Partition

On EFI based systems, the bootloaders are installed in a special FAT32 partition called an *EFI System Partition* (ESP). If your system supports EFI, and a recent version of some Linux distribution or Windows is pre-installed, it's likely that the ESP has already been created. As the `root` user, list all the partitions on your hard drive (replace `sda` with the device corresponding to the appropriate hard drive):

```
fdisk -l /dev/sda
```

The "Type" column of the ESP should be `EFI System`.

If the system or the hard drive is new, or it's the first installation of a UEFI-booted OS on the system, the ESP may not exist. In that case, install [dosfstools-4.2](#) first. Then create a new partition, make a `vfat` file system on it, and set the partition type to "EFI system." See the instructions for the emergency boot device above as a reference.

Warning

Some (old) UEFI implementations may require the ESP to be the first partition on the disk.

Now, as the `root` user, create the mount point for the ESP, and mount it (replace `sda1` with the device node corresponding to the ESP):

```
mount --mkdir -v -t vfat /dev/sda1 -o codepage=437,iocharset=iso8859-1 \
/boot/efi
```

If you want to mount the ESP automatically during system boot, as the `root` user, add an entry for the ESP into `/etc/fstab`:

```
cat >> /etc/fstab << EOF
/dev/sda1 /boot/efi vfat codepage=437,iocharset=iso8859-1 0 1
EOF
```

Minimal Boot Configuration with GRUB and EFI

On UEFI based systems, GRUB works by installing an EFI application (a special kind of executable) into the ESP. The EFI firmware will search boot loaders in EFI applications from boot entries recorded in EFI variables, and additionally a hardcoded path `EFI/BOOT/BOOTX64.EFI`. Normally, a boot loader should be installed into a custom path and the path should be recorded in the EFI variables. The use of the hardcoded path should be avoided if possible. However, in some cases we have to use the hardcoded path:

- The system is not booted with EFI yet, making EFI variables inaccessible.
- The EFI firmware is 64-bit but the LFS system is 32-bit, making EFI variables inaccessible because the kernel cannot invoke EFI runtime services with a different virtual address length.
- LFS is built for a Live USB, so we cannot rely on EFI variables, which are stored in NVRAM or EEPROM on the local machine.
- You are unable or unwilling to install the `efibootmgr` for manipulating boot entries in EFI variables.

In these cases, follow these instructions to install the GRUB EFI application into the hardcoded path and make a minimal boot configuration. Otherwise it's better to skip ahead and set up the boot configuration normally.

To install GRUB with the EFI application in the hardcoded path `EFI/BOOT/BOOTX64.EFI`, first ensure the boot partition is mounted at `/boot` and the ESP is mounted at `/boot/efi`. Then, as the `root` user, run the command:

Note

This command will overwrite `/boot/efi/EFI/BOOT/BOOTX64.EFI`. It may break a bootloader already installed there. Back it up if you are not sure.

```
grub-install --target=x86_64-efi --removable
```

This command will install the GRUB EFI application into the hardcoded path `/boot/efi/EFI/BOOT/BOOTX64.EFI`, so the EFI firmware can find and load it. The remaining GRUB files are installed in the `/boot/grub` directory and will be loaded by `BOOTX64.EFI` during system boot.

Note

The EFI firmware usually prefers the EFI applications with a path stored in EFI variables to the EFI application at the hardcoded path. So you may need to invoke the boot selection menu or firmware setting interface to select the newly installed GRUB manually on the next boot. Read the manual of your motherboard or laptop to learn how.

If you've followed the instructions in this section and set up a minimal boot configuration, now skip ahead to "Creating the GRUB Configuration File."

Mount the EFI Variable File System

The installation of GRUB on a UEFI platform requires that the EFI Variable file system, `efivarfs`, is mounted. As the `root` user, mount it if it's not already mounted:

```
mountpoint /sys/firmware/efi/efivars || mount -v -t efivarfs efivars /sys/firmware/efi/efivars
```

Note

If the system is booted with UEFI and `systemd`, `efivarfs` will be mounted automatically. However, in the LFS chroot environment it still needs to be mounted manually.

Warning

If the system is not booted with UEFI, the directory `/sys/firmware/efi` will be missing. In this case you should boot the system in UEFI mode with the emergency boot disk or using a minimal boot configuration created as above, then mount `efivarfs` and continue.

Setting Up the Configuration

On UEFI based systems, GRUB works by installing an EFI application (a special kind of executable) into `/boot/efi/EFI/[id]/grubx64.efi`, where `/boot/efi` is the mount point of the ESP, and `[id]` is replaced with an identifier specified in the `grub-install` command line. GRUB will create an entry in the EFI variables containing the path `EFI/[id]/grubx64.efi` so the EFI firmware can find `grubx64.efi` and load it.

`grubx64.efi` is very lightweight (136 KB with GRUB-2.06) so it will not use much space in the ESP. A typical ESP size is 100 MB (for Windows boot manager, which uses about 50 MB in the ESP). Once `grubx64.efi` has been loaded by the firmware, it will load GRUB modules from the boot partition. The default location is `/boot/grub`.

As the `root` user, install the GRUB files into `/boot/efi/EFI/LFS/grubx64.efi` and `/boot/grub`. Then set up the boot entry in the EFI variables:

```
grub-install --bootloader-id=LFS --recheck
```

If the installation is successful, the output should be:

```
Installing for x86_64-efi platform.  
Installation finished. No error reported.
```

Issue the `efibootmgr | cut -f 1` command to recheck the EFI boot configuration. An example of the output is:

```
BootCurrent: 0000  
Timeout: 1 seconds  
BootOrder: 0005,0000,0002,0001,0003,0004  
Boot0000* ARCH  
Boot0001* UEFI:CD/DVD Drive  
Boot0002* Windows Boot Manager  
Boot0003* UEFI:Removable Device  
Boot0004* UEFI:Network Device  
Boot0005* LFS
```

Note that `0005` is the first in the `BootOrder`, and `Boot0005` is `LFS`. This means that on the next boot, the version of GRUB installed by `LFS` will be used to boot the system.

Creating the GRUB Configuration File

Generate `/boot/grub/grub.cfg` to configure the boot menu of GRUB:

```
cat > /boot/grub/grub.cfg << EOF  
# Begin /boot/grub/grub.cfg  
set default=0  
set timeout=5  
  
insmod part_gpt  
insmod ext2  
set root=(hd0,2)  
  
insmod efi_gop  
insmod efi_uga  
if loadfont /boot/grub/fonts/unicode.pf2; then  
    terminal_output gfxterm  
fi  
  
menuentry "GNU/Linux, Linux 6.10.5-lfs-12.2" {  
    linux   /boot/vmlinuz-6.10.5-lfs-12.2 root=/dev/sda2 ro  
}  
  
menuentry "Firmware Setup" {  
    fwsetup  
}  
EOF
```

Refer to [the LFS book](#) for the basic knowledge about the `grub.cfg` file. `(hd0,2)`, `sda2`, and `6.10.5-lfs-12.2` must match your configuration.

The `insmod efi_gop` and `insmod efi_uga` directives load two modules for EFI-based video support. On most systems the `efi_gop` module is enough. The `efi_uga` module is only useful for legacy systems, but it's harmless to load it anyway. The video support is needed for the `terminal_output gfxterm` directive to really work.

The `terminal_output gfxterm` directive changes the display resolution of the GRUB menu to match your display device. It will break the rendering if the `unicode.pf2` font data file is not loaded, so it's guarded by a `if` directive.

Note

From GRUB's perspective, the files are relative to the partitions used. If you used a separate `/boot` partition, remove `/boot` from the above paths (to kernel and to `unicode.pf2`). You will also need to change the "set root" line to point to the boot partition.

The `Firmware Setup` entry can be used to enter the configuration interface provided by the firmware (sometimes called "BIOS configuration").

Dual-booting with Windows

Add a menu entry for Windows into `grub.cfg`:

```
cat >> /boot/grub/grub.cfg << EOF
# Begin Windows addition

menuentry "Windows 11" {
    insmod fat
    insmod chain
    set root=(hd0,1)
    chainloader /EFI/Microsoft/Boot/bootmgfw.efs
}
EOF
```

(`hd0,1`) should be replaced with the GRUB designated name for the ESP. The `chainloader` directive can be used to tell GRUB to run another EFI executable, in this case the Windows Boot Manager. You may put more usable tools in EFI executable format (for example, an EFI shell) into the ESP and create GRUB entries for them, as well.

Chapter 6. Text Editors

This chapter is referenced in the LFS book for those wishing to use other editors on their LFS system. You're also shown how some LFS installed programs benefit from being recompiled after GUI libraries have been installed.

Bluefish-2.2.15

Introduction to Bluefish

Bluefish is a GTK+ text editor targeted towards programmers and web designers, with many options to write websites, scripts and programming code. Bluefish supports many programming and markup languages, and it focuses on editing dynamic and interactive websites.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.bennewitz.com/bluefish/stable/source/bluefish-2.2.15.tar.bz2>
- Download MD5 sum: 48018bcfef0faeceb66b0f2d814ded03
- Download size: 4.5 MB
- Estimated disk space required: 52 MB
- Estimated build time: 0.4 SBU

Bluefish Dependencies

Required

[GTK+-3.24.43](#)

Recommended

[desktop-file-utils-0.27](#) (for updating the desktop database)

Optional

[enchant-2.8.2](#) (for spell checking), [Gucharmap-15.1.5](#), and [Jing](#).

Installation of Bluefish

First, disable a plugin requiring functions removed in libxml-2.13.0 or later:

```
sed '/infbrowser/d' -i src/Makefile.am &&
autoreconf
```

Install Bluefish by running the following commands:

```
./configure --prefix=/usr --docdir=/usr/share/doc/bluefish-2.2.15 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/icon-theme.cache` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -t -f --include-image-data /usr/share/icons/hicolor &&
update-desktop-database
```

Contents

Installed Program: bluefish

Installed Libraries: several under `/usr/lib/bluefish/`

Installed Directories: `/usr/lib/bluefish`, `/usr/share/bluefish`, `/usr/share/doc/bluefish-2.2.15`, and `/usr/share/xml/bluefish`

Short Descriptions

`bluefish` is a GTK+ text editor for markup and programming

Ed-1.20.2

Introduction to Ed

Ed is a line-oriented text editor. It is used to create, display, modify and otherwise manipulate text files, both interactively and via shell scripts. Ed isn't something which many people use. It's described here because it can be used by the patch program if you encounter an ed-based patch file. This happens rarely because diff-based patches are preferred these days.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/ed/ed-1.20.2.tar.lz>
- Download MD5 sum: 8f31ca1cb5a2e2ca8ee2598cc0bf53e1
- Download size: 68 KB
- Estimated disk space required: 1.0 MB
- Estimated build time: less than 0.1 SBU

Ed Dependencies

Required to uncompress the tarball

[libarchive-3.7.4](#) (for bsdtar)

Installation of Ed

Install Ed by running the following commands:

```
./configure --prefix=/usr &&  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: ed and red

Installed Libraries: None

Installed Directories: None

Short Descriptions

- `ed` is a line-oriented text editor
- `red` is a restricted `ed`—it can only edit files in the current directory and cannot execute shell commands

Emacs-29.4

Introduction to Emacs

The Emacs package contains an extensible, customizable, self-documenting real-time display editor.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/emacs/emacs-29.4.tar.xz>
- Download MD5 sum: b9cc42f7d8066152535cf267418b8ced
- Download size: 50 MB
- Estimated disk space required: 525 MB
- Estimated build time: 0.7 SBU (Using parallelism=4)

Emacs Dependencies

Recommended

[harfBuzz-9.0.0](#), [giflib-5.2.2](#), [GnuTLS-3.8.7.1](#), [jansson-2.14](#), and [libtiff-4.6.0](#)

Optional

[a graphical environment](#), [alsa-lib-1.2.12](#), [Cairo-1.18.0](#), [dbus-1.14.10](#), [GLib-2.80.4](#) (with GObject Introspection), [gsettings-desktop-schemas-46.1](#), [GPM-1.20.7](#), [GTK+-3.24.43](#), [ImageMagick-7.1.1-36](#), [Little CMS-2.16](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), [librsvg-2.58.3](#), [libseccomp-2.5.5](#), [libwebp-1.4.0](#), [libxml2-2.13.3](#), [MIT Kerberos V5-1.21.3](#), [SQLite-3.46.1](#), [Valgrind-3.23.0](#), [intlfonts](#), [libungif](#), [libotf](#), and [m17n-lib](#) - to correctly display such complex scripts as Indic and Khmer, and also for scripts that require Arabic shaping support (Arabic and Farsi), [mailutils](#), and [libXaw3d](#)

Installation of Emacs

Install Emacs by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite. If make succeeds, you can test the result by running `src/emacs -q`, which is the program that will be installed, with its auxiliary files. This should start and display the application opening screen.

Now, as the `root` user:

```
make install &&
chown -v -R root:root /usr/share/emacs/29.4
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and you can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme`. To perform the update you must have [GTK+-3.24.43](#) installed and issue the following command as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor
```

Command Explanations

--with-imagemagick: Use this if you have installed [ImageMagick-7.1.1-36](#) and wish to link against it.
--with-gif=no: Use this if you have not installed [giflib-5.2.2](#) or [libungif](#).
--with-tiff=no: Use this if you have not installed [ibtiff-4.6.0](#).
--with-gnutls=no: Use this if you have not installed [GnuTLS-3.8.7.1](#).
--without-harfbuzz: Use this if you have not installed [harfBuzz-9.0.0](#).
--with-json=no: Use this if you have not installed [jansson-2.14](#).

Contents

Installed Programs: ctags, ebrowse, emacs (symlink), emacs-29.4, emacsclient, and etags

Installed Libraries: None

Installed Directories: /usr/libexec/emacs and /usr/share/emacs

Short Descriptions

<code>ctags</code>	creates cross-reference tagfile database files for source code
<code>ebrowse</code>	permits browsing of C++ class hierarchies from within <code>emacs</code>
<code>emacs</code>	is an editor
<code>emacsclient</code>	attaches an <code>emacs</code> session to an already running <code>emacs</code> instance
<code>etags</code>	is another program to generate source code cross-reference tagfiles

Gedit-47.0

Introduction to Gedit

The Gedit package contains a lightweight UTF-8 text editor for the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gedit/47/gedit-47.0.tar.xz>
- Download MD5 sum: 88aa416009439007815a3c01ce45e4fe
- Download size: 2.7 MB
- Estimated disk space required: 66 MB (with tests)
- Estimated build time: 0.2 SBU (using parallelism=4; with tests)

Additional Downloads

- Download (HTTP): <https://github.com/gedit-technology/libgedit-tepl/releases/download/6.10.0/libgedit-tepl-6.10.0.tar.xz>
- Download MD5 sum: 0bc684817b93abbcb290094bc94abdb
- Download size: 168 KB
- Estimated disk space required: 10 MB (with tests)
- Estimated build time: 0.3 SBU (with tests)

- Download (HTTP): <https://github.com/gedit-technology/libgedit-gfls/releases/download/0.1.0/libgedit-gfls-0.1.0.tar.xz>
- Download MD5 sum: fa465c84be20de668b00deb2723334f3
- Download size: 20 KB
- Estimated disk space required: 1.0 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Gedit Dependencies

Required

[gsettings-desktop-schemas-46.1](#), [itstool-2.0.7](#), [libgedit-amtk-5.8.0](#), [libgeditsourceview-299.2.1](#), [libhandy-1.8.3](#), and [libpeas-1.36.0](#)

Recommended

[gspell-1.12.2](#), [Gvfs-1.54.2](#) (runtime), [ISO Codes-4.16.0](#), and [PyGObject-3.48.2](#) (Python3 module)

Optional

[GTK-Doc-1.34.0](#), [Vala-0.56.17](#), and [zeitgeist](#)

Installation of Gedit

First, build the gfls library:

```
tar -xf ../libgedit-gfls-0.1.0.tar.xz &&
pushd libgedit-gfls-0.1.0 &&

mkdir gfls-build &&
cd gfls-build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D gtk_doc=false \
            ..
            &&
ninja
```

To test the results, run: `ninja test`.

Now, as the `root` user:

```
ninja install &&
popd
```

Second, build the tepl library:

```
tar -xf ../libgedit-tepl-6.10.0.tar.xz &&
pushd libgedit-tepl-6.10.0 &&

mkdir tepl-build &&
cd tepl-build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D gtk_doc=false \
            ..
            &&
ninja
```

To test the results, run: `ninja test`.

Now, as the `root` user:

```
ninja install &&
popd
```

Finally, install Gedit by running the following commands:

```
mkdir gedit-build &&
cd gedit-build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false    \
            ... &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a “`DESTDIR`” method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Command Explanations

`-D gtk_doc=false`: This switch disables generating the API documentation. Omit this switch if you have [GTK-Doc-1.34.0](#) installed and wish to generate the API documentation.

Contents

Installed Program: gedit

Installed Libraries: libgedit-gfls-1.so, libgedit-tepl-6.so, and libgedit-46.so

Installed Directories: /usr/include/libgedit-gfls-1, /usr/include/libgedit-tepl-6, /usr/include/gedit-46, and /usr/{lib,share,share/help/*}/gedit

Short Descriptions

`gedit` is a lightweight text editor integrated with the GNOME Desktop

JOE-4.6

Introduction to JOE

JOE (Joe's own editor) is a small text editor capable of emulating WordStar, Pico, and Emacs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/joe-editor/joe-4.6.tar.gz>
- Download MD5 sum: 9017484e6116830d846678b625ea5c43
- Download size: 1.8 MB
- Estimated disk space required: 22 MB
- Estimated build time: 0.2 SBU

Installation of JOE

Install JOE by running the following commands:

```
./configure --prefix=/usr \
--sysconfdir=/etc \
--docdir=/usr/share/doc/joe-4.6 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -vm 755 joe/util/{stringify,termidx,uniproc} /usr/bin
```

Configuring JOE

Config Files

`/etc/joe/jmacsrc`, `/etc/joe/joerc`, `/etc/joe/jpicorc`, `/etc/joe/jstarrc`, `/etc/joe/rjoerc`, and `~/.joerc`

Contents

Installed Programs: `jmacs`, `joe`, `jpico`, `jstar`, `rjoe`, `stringify`, `termidx`, and `uniproc`

Installed Libraries: None

Installed Directories: `/etc/joe`, `/usr/share/joe`, and `/usr/share/doc/joe-4.6`

Short Descriptions

<code>jmacs</code>	is a symbolic link to <code>joe</code> used to launch Emacs emulation mode
<code>joe</code>	is a small text editor capable of emulating WordStar, Pico , and Emacs
<code>jpico</code>	is a symbolic link to <code>joe</code> used to launch Pico emulation mode
<code>jstar</code>	is a symbolic link to <code>joe</code> used to launch WordStar emulation mode
<code>rjoe</code>	is a symbolic link to <code>joe</code> that restricts JOE to editing only files which are specified on the command-line
<code>stringify</code>	is a program used by <code>joe</code> to convert rc and .jsf files into a C file (see <code>/usr/share/doc/joe-4.6/util/README</code>)
<code>termidx</code>	is a program used by <code>joe</code> to generate the termcap index file (see <code>/usr/share/doc/joe-4.6/util/README</code>)
<code>uniproc</code>	is a program used by <code>joe</code> to generate joe's unicode database file unicat.c from Blocks.txt CaseFolding.txt EastAsianWidth.txt and UnicodeData.txt (find them at <code>/usr/share/doc/joe-4.6/util</code> ; see <code>usr/share/doc/joe-4.6/util/README</code>)

kate-24.08.0

Introduction to Kate

The Kate package contains an advanced KF6 based graphical text editor.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/kate-24.08.0.tar.xz>
- Download MD5 sum: `bdb43ea4ec735e3a9c81701257581ad6`
- Download size: 7.8 MB
- Estimated disk space required: 299 MB
- Estimated build time: 1.0 SBU (using parallelism=4)

Kate Dependencies

Required

Installation of Kate

Install Kate by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `kate` and `kwrite`

Installed Libraries: Several plugins under `$KF6_PREFIX/lib/plugins`

Installed Directories: `$KF6_PREFIX/lib/plugins/kttexteditor`, `$KF6_PREFIX/lib/plugins/plasma/dataengine`,
`$KF6_PREFIX/share/doc/HTML/*/{kate,katepart,kwrite}`,
`$KF6_PREFIX/share/{kateproject,katexmltools}`, and
`$KF6_PREFIX/share/plasma/plasmoids/org.kde.plasma.katesessions`

Short Descriptions

`kate` is an advanced text editor for kde
`kwrite` is a text editor for KDE, that is a light version of `kate`

Mousepad-0.6.2

Introduction to Mousepad

Mousepad is a simple GTK+ 3 text editor for the Xfce desktop environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/apps/mousepad/0.6/mousepad-0.6.2.tar.bz2>
- Download MD5 sum: 3ad46198202d2696cac27d5a0f08bab0
- Download size: 1.3 MB
- Estimated disk space required: 16 MB
- Estimated build time: 0.1 SBU

Mousepad Dependencies

Required

[gtksourceview4-4.8.4](#)

Optional

[DConf-0.40.0](#) (runtime) and [dbus-glib-0.112](#)

Installation of Mousepad

Install Mousepad by running the following commands:

```
./configure --prefix=/usr      \
--enable-gtksourceview4 \
--enable-keyfile-settings &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-keyfile-settings`: Use the GSettings keyfile backend rather than the default [DConf-0.40.0](#).

Contents

Installed Program: mousepad

Installed Libraries: None

Installed Directories: None

Short Descriptions

`mousepad` is a simple GTK+ 3 text editor

Nano-8.1

Introduction to Nano

The Nano package contains a small, simple text editor which aims to replace Pico, the default editor in the Pine package.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.nano-editor.org/dist/v8/nano-8.1.tar.xz>
- Download MD5 sum: 9eb581845590ad70ba89c04509c7a386
- Download size: 1.6 MB
- Estimated disk space required: 23 MB
- Estimated build time: 0.2 SBU

Installation of Nano

Install Nano by running the following commands:

```
./configure --prefix=/usr      \
--sysconfdir=/etc \
--enable-utf8 \
--docdir=/usr/share/doc/nano-8.1 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -v -m644 doc/{nano.html,sample.nanorc} /usr/share/doc/nano-8.1
```

Command Explanations

`--enable-utf8`: This switch enables unicode support in Nano.

Configuring nano

Config Files

/etc/nanorc and ~/.nanorc

Configuration Information

Example configuration (create as a system-wide /etc/nanorc or a personal ~/.nanorc file)

```
set autoindent
set constantshow
set fill 72
set historylog
set multibuffer
set nohelp
set positionlog
set quickblank
set regexp
```

Check the `sample.nanorc` file in the installed documentation directory. It includes color configurations and has some documentation included in the comments.

Syntax highlighting is provided for several file types, in `/usr/share/nano/` directory. E.g., for shell scripts, you can insert `include /usr/share/nano/sh.nanorc` in the personal or global configuration file. If you wish highlighting for all supported files, use `include /usr/share/nano/*.nanorc`. This include does not descend into the `extra` directory. Move required files one level up.

Contents

Installed Programs: nano and rnano (symlink)

Installed Libraries: None

Installed Directories: /usr/share/nano and /usr/share/doc/nano-8.1

Short Descriptions

`nano` is a small, simple text editor which aims to replace Pico , the default editor in the Pine package
`rnano` is a restricted mode for `nano`

Vim-9.1.0660

Introduction to Vim

The Vim package, which is an abbreviation for VI IMproved, contains a `vi` clone with extra features as compared to the original `vi`.

The default LFS instructions install vim as a part of the base system. If you would prefer to link vim against X, you should recompile vim to enable GUI mode. There is no need for special instructions since X support is automatically detected.

Note

The version of vim changes daily. To get the latest version, go to <https://github.com/vim/vim/releases>.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/vim/vim/archive/v9.1.0660/vim-9.1.0660.tar.gz>
- Download MD5 sum: c512a99b3704f193be1a181cc644b2b2
- Download size: 17 MB
- Estimated disk space required: 134 MB (add 74 MB for tests)
- Estimated build time: 0.3 SBU (with parallelism=4; add 11 SBU for tests)

Vim Dependencies

Recommended

[a graphical environment](#) and [GTK+-3.24.43](#)

Optional

[GPM-1.20.7](#), [Lua-5.4.7](#), [Ruby-3.3.4](#), and [rsync-3.3.0](#)

Installation of Vim

Note

If you recompile Vim to link against X and your X libraries are not on the root partition, you will no longer have an editor for use in emergencies. You may choose to install an additional editor, not link Vim against X, or move the current `vim` executable to the `/bin` directory under a different name such as `vi`.

Install Vim by running the following commands:

Note

If you intend to run the tests and have not installed Xorg in `/usr`, append `LDFLAGS=-L$XORG_PREFIX/lib` to the configure line below.

```
echo '#define SYS_VIMRC_FILE "/etc/vimrc"' >> src/feature.h &&
echo '#define SYS_GVIMRC_FILE "/etc/gvimrc"' >> src/feature.h &&

./configure --prefix=/usr      \
            --with-features=huge \
            --enable-gui=gtk3   \
            --with-tlib=ncursesw &&
make
```

Note

If the global configuration file `/etc/vimrc` references the `VIMRUNTIME` environment variable, some tests may complain about being unable to find the corresponding directory and wait for user input. If this is the case, this file should be saved and removed before running the tests. Or if [bubblewrap-0.9.0](#) is installed, it's also possible to create a lightweight container environment where this file is hidden and run the tests in the container.

To test the results, issue: `make -j1 test` Test failures, if any, will produce the file `test.log` in `src/testdir`. The remaining tests will still be executed. If all goes well, the log will report `ALL DONE`. Some tests labelled as "flaky" may fail occasionally and can be ignored. The tests are known to fail if the output is redirected to a file, and also if they are run in a 'screen' session.

Note

If running the tests with [bubblewrap-0.9.0](#) and `/etc/vimrc` hidden, use `bwrap --dev-bind / / --dev-bind /dev/null /etc/vimrc make -j1 test`.

Note

Some color tests expect to be executed under the `xterm` terminal emulator.

Now, as the `root` user:

```
make install
```

By default, Vim's documentation is installed in `/usr/share/vim`. The following symlink allows the documentation to be accessed via `/usr/share/doc/vim-9.1.0660`, making it consistent with the location of documentation for other packages:

```
ln -snfv ../../vim/vim91/doc /usr/share/doc/vim-9.1.0660
```

If you wish to update the runtime files, issue the following command (requires [rsync-3.3.0](#)) to download it:

```
rsync -avzcP --exclude="/dos/" --exclude="/spell/" \
      ftp.nluug.nl::Vim/runtime/ ./runtime/
```

And then install the updated runtime files and regenerate the `tags` file, as the `root` user issue:

```
make -C src installruntime &&
vim -c ":helptags /usr/share/doc/vim-9.1.0660" -c ":q"
```

Command Explanations

`--with-features=huge`: This switch enables all the additional features available in Vim, including support for multibyte characters.

`--with-tlib=ncursesw`: This switch forces Vim to link against the `libncursesw` library.

`--enable-gui=no`: This will prevent compilation of the GUI. Vim will still link against X, so that some features such as the client-server model or the x11-selection (clipboard) are still available.

`--without-x`: If you prefer not to link Vim against X, use this switch.

`--enable-luainterp`, `--enable-perlinterp`, `--enable-python3interp=dynamic`, `--enable-tclinterp` --with-tclsh=tclsh, `--enable-rubyinterp`: These options include the Lua, Perl, Python3, Tcl, or Ruby interpreters that allow using other application code in vim scripts. All the `--enable-*` options can accept `=dynamic` to dynamically load the interpreter when needed. This is required for Python 3 to prevent segmentation faults. For tcl, it is necessary to indicate the name of the `tclsh` executable, since `configure` only searches versioned names with old versions.

Configuring Vim

Config Files

`/etc/vimrc` and `~/.vimrc`

Configuration Information

Vim has an integrated spell checker which you can enable by issuing the following in a vim window:

```
:setlocal spell spelllang=ru
```

This setting will enable spell checking for the Russian language for the current session.

By default, Vim only installs spell files for the English language. If a spell file is not available for a language, then Vim will call the `$VIMRUNTIME/plugin/spellfile.vim` plugin and will try to obtain the `*.spl` and optionally `*.sug` from the vim ftp server, by using the `$VIMRUNTIME/plugin/netrwPlugin.vim` plugin.

Alternatively you can manually download the `*.spl` and `*.sug` files from: <ftp://ftp.vim.org/pub/vim/runtime/spell/> and save them to `~/.vim/spell` or in `/usr/share/vim/vim91/spell/`.

To find out what's new in Vim-9.1.0660 issue the following command:

```
:help version-9.1.0660
```

For additional information on setting up Vim configuration files, see [The vimrc Files](#) and https://vim.fandom.com/wiki/Example_vimrc.

Contents

A list of the reinstalled files, along with their short descriptions can be found in the [LFS Vim Installation Instructions](#).

Installed Programs: gview, gvim, gvimdiff, gvimtutor, rgview, and rgvim

Installed Libraries: None

Installed Directory: /usr/share/vim

Short Descriptions

<code>gview</code>	starts <code>gvim</code> in read-only mode
<code>gvim</code>	is the editor that runs under X and includes a GUI

<code>gvimdiff</code>	edits two or three versions of a file with <code>gvim</code> and shows the differences
<code>gvimtutor</code>	teaches the basic keys and commands of <code>gvim</code>
<code>rgview</code>	is a restricted version of <code>gview</code>
<code>rgvim</code>	is a restricted version of <code>gvim</code>

Chapter 7. Shells

We are all familiar with the Bourne Again SHell, but there are two other user interfaces that are considered useful modern shells – the Berkeley Unix C shell and the Korn shell. This chapter installs packages compatible with these additional shell types.

Dash-0.5.12

Introduction to Dash

Dash is a POSIX compliant shell. It can be installed as `/bin/sh` or as the default shell for either `root` or a second user with a userid of 0. It depends on fewer libraries than the Bash shell and is therefore less likely to be affected by an upgrade problem or disk failure. Dash is also useful for checking that a script is completely compatible with POSIX syntax.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://gondor.apana.org.au/~herbert/dash/files/dash-0.5.12.tar.gz>
- Download MD5 sum: 57222b768b84003ea4b801e5d5e0e52b
- Download size: 244 KB
- Estimated disk space required: 2.9 MB
- Estimated build time: less than 0.1 SBU

Dash Dependencies

Optional

[libedit](#) (command line editor library)

Installation of Dash

Install Dash by running the following commands:

```
./configure --bindir=/bin --mandir=/usr/share/man &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you would like to make `dash` the default `sh`, recreate the `/bin/sh` symlink as the `root` user:

Note

If you create the symbolic link from `dash` to `sh`, you will need to reset the link to `bash` to build LFS.

```
ln -svf dash /bin/sh
```

Command Explanations

`--bindir=/bin`: This parameter places the `dash` binary into the root filesystem.

--with-libedit: To compile Dash with libedit support.

Configuring Dash

Config Files

Dash sources `/etc/profile` and `~/.profile`

Configuration Information

Update `/etc/shells` to include the Dash shell by issuing the following command as the `root` user:

```
cat >> /etc/shells << "EOF"
/bin/dash
EOF
```

Contents

Installed Program: dash

Installed Libraries: None

Installed Directories: None

Short Description

`dash` is a POSIX compliant shell

Tcsh-6.24.13

Introduction to Tcsh

The Tcsh package contains “an enhanced but completely compatible version of the Berkeley Unix C shell (`csh`).” This is useful as an alternative shell for those who prefer C syntax to that of the `bash` shell, and also because some programs require the C shell in order to perform installation tasks.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://astron.com/pub/tcsh/tcsh-6.24.13.tar.gz>
- Download MD5 sum: 7467f2a08b758d3439aad332322e6a98
- Download size: 940 KB
- Estimated disk space required: 11 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Installation of Tcsh

Install Tcsh by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install install.man &&
ln -v -sf tcsh    /bin/csh &&
ln -v -sf tcsh.1  /usr/share/man/man1/csh.1
```

Command Explanations

`ln -v -sf tcsh /bin/csh`: The FHS states that if there is a C shell installed, there should be a symlink from `/bin/csh` to it. This creates that symlink.

Configuring Tcsh

Config Files

There are numerous configuration files for the C shell. Examples of these are `/etc/csh.cshrc`, `/etc/csh.login`, `/etc/csh.logout`, `~/.tcshrc`, `~/.cshrc`, `~/.history`, `~/.cshdirs`, `~/.login`, and `~/.logout`. More information on these files can be found in the [tcsh\(1\)](#) man page.

Configuration Information

Update `/etc/shells` to include the C shell program names (as the `root` user):

```
cat >> /etc/shells << "EOF"
/bin/tcsh
/bin/csh
EOF
```

The following `~/.cshrc` provides two alternative colour prompts and coloured `ls` output. If you prefer a global modification, issue the command as the `root` user, replacing `~/.cshrc` by `/etc/csh.cshrc`.

```
cat > ~/.cshrc << "EOF"
# Original at:
# https://www.cs.umd.edu/~srhuang/teaching/code_snippets/prompt_color.tcsh.html

# Modified by the BLFS Development Team.

# Add these lines to your ~/.cshrc (or to /etc/csh.cshrc).

# Colors!
set      red="%{\e[1;31m}"
set      green="%{\e[0;32m}"
set      yellow="%{\e[1;33m}"
set      blue="%{\e[1;34m}"
set magenta="%{\e[1;35m}"
set      cyan="%{\e[1;36m}"
set      white="%{\e[0;37m}"
set      end="%{\e[0m}" # This is needed at the end...

# Setting the actual prompt. Two separate versions for you to try, pick
# whichever one you like better, and change the colors as you want.
# Just don't mess with the ${end} guy in either line... Comment out or
# delete the prompt you don't use.

set prompt="${green}%n${blue}@%m ${white}%~ ${green}%%${end} "
set prompt="[$green%n${blue}@%m ${white}%~ ]${end} "

# This was not in the original URL above
# Provides coloured ls
alias ls ls --color=always

# Clean up after ourselves...
unset red green yellow blue magenta cyan yellow white end
EOF
```

Contents

Installed Program: tcsh

Installed Libraries: None

Installed Directory: None

Short Descriptions

`tcsh` is an enhanced but completely compatible version of the Berkeley Unix C shell, `csh`. It is usable as both an interactive shell and a script processor

Introduction to zsh

The zsh package contains a command interpreter (shell) usable as an interactive login shell and as a shell script command processor. Of the standard shells, zsh most closely resembles ksh but includes many enhancements.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.zsh.org/pub/zsh-5.9.tar.xz>
- Download MD5 sum: 182e37ca3fe3fa6a44f69ad462c5c30e
- Download size: 3.2 MB
- Estimated disk space required: 48 MB (includes documentation and tests)
- Estimated build time: 1.6 SBU (Using parallelism=4; includes documentation and tests)

Note

When there is a new zsh release, the old files shown above are moved to a new server directory:
<https://www.zsh.org/pub/old/>.

zsh Dependencies

Optional

[Valgrind-3.23.0](#) and [yodl](#)

Installation of zsh

Adapt the documentation build system for texinfo-7.0 or later:

```
sed -e 's/set_from_init_file/texinfo_&/' \
      -i Doc/Makefile.in
```

Some programs shipped in the building system for detecting system features use pre-C99 syntax rejected by GCC-14.1 or later. Fix them up or Zsh would be wrongly configured and fail to build:

```
sed -e 's/^main/int &/' \
      -e 's/exit(/return(/' \
      -i aczsh.m4 configure.ac &&

sed -e 's/test = /&(char**)/' \
      -i configure.ac &&

autoconf
```

The documentation files contain references to zsh configuration files in `/etc`, but we'll use `/etc/zsh` to hold these configuration files instead. The building system will fix up those references if the yodl package is available, but it's out of the scope of BLFS. So we need to fix up the references manually:

```
sed -e 's|/etc/zsh|/etc/zsh/zig|' \
      -i Doc/*.*
```

Install zsh by running the following commands:

```
./configure --prefix=/usr \
            --sysconfdir=/etc/zsh \
            --enable-etcdir=/etc/zsh \
            --enable-cap \
            --enable-gdbm &&
make &&
makeinfo Doc/zsh.texi --html -o Doc/html &&
```

```
makeinfo Doc/zsh.texi --plaintext -o zsh.txt &&
makeinfo Doc/zsh.texi --html --no-split --no-headers -o zsh.html
```

If you have [texlive-20240312](#) installed, you can build PDF format of the documentation by issuing the following command:

```
texi2pdf Doc/zsh.texi -o Doc/zsh.pdf
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
make infodir=/usr/share/info install.info &&
make htmldir=/usr/share/doc/zsh-5.9/html install.html &&
install -v -m644 zsh.{html,txt} Etc/FAQ /usr/share/doc/zsh-5.9
```

If you built the PDF format of the documentation, install it by issuing the following command as the `root` user:

```
install -v -m644 Doc/zsh.pdf /usr/share/doc/zsh-5.9
```

Command Explanations

`--sysconfdir=/etc/zsh` and `--enable-etcdir=/etc/zsh`: These parameters are used so that all the zsh configuration files are consolidated into the `/etc/zsh` directory. Omit these parameters if you wish to retain historical compatibility by having all the files located in the `/etc` directory.

`--enable-cap`: This option enables POSIX capabilities.

`--enable-gdbm`: This option enables the use of the GDBM library.

Configuring zsh

Config Files

There are a whole host of configuration files for zsh including `/etc/zsh/zshenv`, `/etc/zsh/zprofile`, `/etc/zsh/zshrc`, `/etc/zsh/zlogin` and `/etc/zsh/zlogout`. You can find more information on these in the [zsh\(1\)](#) and related manual pages.

The first time zsh is executed, you will be prompted by messages asking several questions. The answers will be used to create a `~/.zshrc` file. If you wish to run these questions again, run `zsh /usr/share/zsh/5.9/functions/zsh-newuser-install -f`.

There are several built-in advanced prompts. In the `zsh` shell, start advanced prompt support with `autoload -U promptinit`, then `promptinit`. Available prompt names are listed with `prompt -l`. Select a particular one with `prompt <prompt-name>`. Display all available prompts with `prompt -p`. Except for the list and display commands above, you can insert the other ones in `~/.zshrc` to be automatically executed at shell start, with the prompt you chose.

Configuration Information

Update `/etc/shells` to include the zsh shell program names (as the `root` user):

```
cat >> /etc/shells << "EOF"
/bin/zsh
EOF
```

Contents

Installed Programs: zsh and zsh-5.9 (hardlinked to each other)

Installed Libraries: Numerous plugin helper modules under `/usr/lib/zsh/5.9/`

Installed Directories: `/usr/{lib,share}/zsh` and `/usr/share/doc/zsh-5.9`

Short Description

`zsh` is a shell which has command-line editing, built-in spelling correction, programmable command completion, shell functions (with autoloading), a history mechanism, and a host of other features

Virtualization allows running a complete operating system, or virtual machine (VM), within another operating environment as a task. There are several commercial and open source environments that either emulate another processor or utilize the hardware virtualization features of the host processor.

qemu-9.0.2

Introduction to qemu

qemu is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.qemu.org/qemu-9.0.2.tar.xz>
- Download MD5 sum: f7f0462262d2571f146c6a8adda33b29
- Download size: 126 MB
- Estimated disk space required: 2.2 GB (374 MB installed)
- Estimated build time: 1.6 SBU (add 1.0 SBU for tests, both using parallelism=4)

Qemu Dependencies

Required

[GLib-2.80.4](#) and [Pixman-0.43.4](#)

Recommended

[alsa-lib-1.2.12](#), [dtc-1.7.1](#), [libslirp-4.8.0](#), and [SDL2-2.30.6](#)

Note

If [dtc-1.7.1](#) is not installed, the building system will attempt to download a copy of dtc source code from the Internet.

Optional

[pipewire-1.2.3](#) or [PulseAudio-17.0](#) (can be used instead of alsa-lib), [BlueZ-5.77](#), [cURL-8.9.1](#), [Cyrus SASL-2.1.28](#), [Fuse-3.16.2](#), [GnuTLS-3.8.7.1](#), [GTK+-3.24.43](#), [keyutils-1.6.3](#), [libaio-0.3.113](#), [libusb-1.0.27](#), [libgcrypt-1.11.0](#), [libjpeg-turbo-3.0.1](#), [libseccomp-2.5.5](#), [libssh2-1.11.0](#), [libpng-1.6.43](#), [libtasn1-4.19.0](#), [Linux-PAM-1.6.1](#), [LZO-2.10](#), [Nettle-3.10](#), [Mesa-24.1.5](#), [VTE-0.76.4](#), [capstone](#), [ceph](#), [daxctl](#), [JACK](#), [glusterfs](#), [libbpf](#), [libcacard](#), [libcap-ng](#), [libdw](#), [libiscsi](#), [libnfs](#), [libpmem](#), [libssh](#), [libu2f-emu](#), [lzfse](#), [netmap](#), [numactl](#), [rdma-core](#), [SELinux](#), [snappy](#), [spice](#), [usbredir](#), and [VDE](#)

Optional (Runtime)

[Systemd-256.4](#)

Optional (for building the documentation)

[sphinx_rtd_theme-2.0.0](#)

Note

This optional dependencies list is not comprehensive. See the output of `./configure --help` for a more complete list.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/qemu>

KVM Prerequisites

Before building qemu, check to see if your processor supports Virtualization Technology (VT):

```
grep -E '^flags.*(vmx|svm)' /proc/cpuinfo
```

If you get any output, you have VT technology (vmx for Intel processors and svm for AMD processors). You then need to go into your system BIOS and ensure it is enabled. After enabling, reboot back to your LFS instance.

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
[*] Virtualization ---> [VIRTUALIZATION]
  <*/M>   Kernel-based Virtual Machine (KVM) support [KVM]
  # Enable the option for your CPU:
  < /*/M>   KVM for Intel (and compatible) processors support [KVM_INTEL]
  < /*/M>   KVM for AMD processors support [KVM_AMD]
```

The Intel or AMD settings are not both required, but the one matching your system processor is required.

To use the “bridge” network device, as explained below, check that [bridge-utils-1.7.1](#) is installed and the following options in the kernel configuration are enabled:

```
[*] Networking support ---> [NET]
  Networking options --->
    <*/M> 802.1d Ethernet Bridging [BRIDGE]

Device Drivers --->
  [*] Network device support ---> [NETDEVICES]
    [*] Network core driver support [NET_CORE]
    <*/M> Universal TUN/TAP device driver support [TUN]
```

Installation of qemu

The udev rule of LFS only allows the `root` user, the users owning a local login session supported by the optional runtime dependency [Systemd-256.4](#), or the users in the `kvm` group to use the KVM device. As the `root` user, add any non-`root` users that might use the KVM device either without [Systemd-256.4](#) installed or remotely (via a SSH connection) to the `kvm` group:

```
usermod -a -G kvm <username>
```

Install qemu by running the following commands:

Note

Qemu is capable of running many targets. The build process is also capable of building multiple targets at one time in a comma delimited list assigned to `--target-list`. Run `./configure --help` to get a complete list of available targets.

```
if [ $(uname -m) = i686 ]; then
  QEMU_ARCH=i386-softmmu
else
  QEMU_ARCH=x86_64-softmmu
fi

mkdir -vp build &&
cd      build &&

./configure --prefix=/usr          \
            --sysconfdir=/etc        \
            --localstatedir=/var       \
            --target-list=$QEMU_ARCH  \
            --audio-drv-list=alsa     \
            --disable-pa              \
            --enable-slirp            \
            --docdir=/usr/share/doc/qemu-9.0.2 &&

unset QEMU_ARCH &&
make
```

qemu uses `ninja` as a subprocess when building. To run the tests, issue: `ninja test`. One test, bios-tables-test, is known to fail.

Now, as the `root` user:

```
make install
```

Change the permissions and ownership of a helper script, which is needed when using the “bridge” network device (see below). Again as the `root` user, issue:

Note

You need to add any users who might use the “bridge” network device into the `kvm` group even if [Systemd-256.4](#) is installed.

```
chgrp kvm /usr/libexec/qemu-bridge-helper &&
chmod 4750 /usr/libexec/qemu-bridge-helper
```

Note

For convenience you may want to create a symbolic link to run the installed program. For instance (as the `root` user):

```
ln -sv qemu-system-`uname -m` /usr/bin/qemu
```

Command Explanations

`--audio-drv-list=alsa`: This switch sets the audio driver to ALSA. See below for enabling other audio drivers.

`--disable-pa`: even if `pa` is not in `--audio-drv-list` list, the pulseaudio driver is built, unless disabled by this parameter.

`--enable-slirp`: This switch forces the building system to check for [libsirp-4.8.0](#). Remove it if you don't need the `-netdev` user support.

`--audio-drv-list=pa --disable-alsa`: This switch sets the audio driver to pulseaudio. For other drivers see the `--audio-drv-list` choices in the output of `./configure --help`. The default audio driver is OSS. To enable support for both alsal and pulseaudio, use `--audio-drv-list=alsa,pa`.

Using Qemu

Since using qemu means using a virtual computer, the steps to set up the virtual machine are in close analogy with those to set up a real computer. You'll need to decide about CPU, memory, disk, USB devices, network card(s), screen size, etc. Once the “hardware” is decided, you'll have for example to choose how to connect the machine to internet, and/or to install an OS. In the following, we show basic ways of performing those steps. But qemu is much more than this, and it is strongly advised to read the qemu documentation in `/usr/share/doc/qemu-9.0.2/qemu-doc.html`.

Note

It is standard practice to name the computer running qemu “host” and the emulated machine running under qemu the “guest.” We'll use those notations in the following.

Note

The following instructions assume the optional symbolic link, `qemu`, has been created. Additionally, `qemu` should be run in a graphical environment. But it is possible to use qemu “headless” or through SSH. See the documentation for the various possibilities.

Disk

A virtual disk may be set up in the following way:

```
VDISK_SIZE=50G
VDISK_FILENAME=vdisk.img
qemu-img create -f qcow2 $VDISK_FILENAME $VDISK_SIZE
```

The virtual disk size and filename should be adjusted as desired. The actual size of the file will be less than specified, but will expand as needed, so it is safe to put a high value.

Operating System

To install an operating system, download an iso image from your preferred Linux distribution. For the purposes of this example, we'll use `Fedora-16-x86_64-Live-LXDE.iso` in the current directory. Run the following:

```
qemu -enable-kvm \
      -drive file=$VDISK_FILENAME \
      -cdrom Fedora-16-x86_64-Live-LXDE.iso \
      -boot d \
      -m 1G
```

Follow the normal installation procedures for the chosen distribution. The `-boot` option specifies the boot order of drives as a string of drive letters. Valid drive letters are: a, b (floppy 1 and 2), c (first hard disk), d (first CD-ROM). The `-m` option is the amount of memory to use for the virtual machine. The choice depends on the load of the host. Modern distributions should be comfortable with 1GB. The `-enable-kvm` option allows hardware acceleration. Without this switch, the emulation is much slower.

Defining the virtual hardware

The virtual machine hardware is defined by the `qemu` command line. An example command is given below:

```
qemu -enable-kvm \
      -smp 4 \
      -cpu host \
      -m 1G \
      -drive file=$VDISK_FILENAME \
      -cdrom grub-img.iso \
      -boot order=c,once=d,menu=on \
      -net nic,netdev=net0 \
      -netdev user,id=net0 \
      -device ac97 \
      -vga std \
      -serial mon:stdio \
      -name "fedora-16"
```

Meaning of the command line options

`-enable-kvm`: enable full KVM virtualization support. On some hardware, it may be necessary to add the undocumented `-machine smm=off` option in order to enable KVM.

`-smp <N>`: enable symmetric multiprocessing with `<N>` CPUs.

`-cpu <model>`: simulate CPU `<model>`. the list of supported models can be obtained with `-cpu help`.

`-drive file=<filename>`: defines a virtual disk whose image is stored in `<filename>`.

`-cdrom grub-img.iso`: defines an iso formatted file to use as a cdrom. Here we use a grub rescue disk, which may turn handy when something goes wrong at boot time.

`-boot order=c,once=d,menu=on`: defines the boot order for the virtual BIOS.

`-net nic,netdev=<netid>`: defines a network card connected to the network device with id `<netid>`.

`-netdev user,id=<netid>`: defines the network "user" device. This is a virtual local network with addresses 10.0.2.0/24, where the host has address 10.0.2.2 and acts as a gateway to internet, and with a name server at address 10.0.2.3, and an smb server at address 10.0.2.4. A builtin DHCP server can allocate addresses between 10.0.2.15 and 10.0.2.31.

`-soundhw <model>`: defines the soundcard model. The list may be obtained with `-soundhw help`.

`-vga <type>`: defines the type of VGA card to emulate. For `-vga std`, if you are building a Linux kernel for the guest, it's recommended to enable `CONFIG_DRM_BOCHS` (as a part of the kernel or a kernel module) to drive all the features of the emulated VGA card, and `CONFIG_FB` to display the Linux console on it. The other `<type>` values are not tested by the editors and may require additional dependencies.

`-serial mon:stdio`: sends the serial port of the guest (`/dev/ttys0` on linux guests), multiplexed with the `qemu` monitor, to the standard input and output of the `qemu` process.

`-name <name>`: sets the name of the guest. This name is displayed in the guest window caption. It may be useful if you run several guests at the same time.

`-drive if=pflash,format=raw,readonly=on,file=/usr/share/qemu/edk2-x86_64-code.fd`: Load a pre-built EDK2 UEFI firmware, instead of the default PC BIOS. Use this option if you want to boot the guest OS with UEFI.

`-drive file=<filename>,if=virtio`: Provide Virtio interface to the guest kernel for accessing the disk image, instead of simulating a real disk hardware. This can improve disk I/O performance, but it requires a Virtio driver in guest kernel. Use it instead of a plain `-drive` if the guest kernel supports Virtio. To build a Linux kernel with Virtio support for the guest, use `make defconfig && make kvm_guest.config` to create an initial kernel configuration with the Virtio drives enabled, then make your customization. And, if the guest kernel is Linux, the virtual disks using Virtio interface will be named `vdx` in the devtmpfs, instead of `sdx`.

`-net nic,netdev=net0,model=virtio-net-pci`: Provide Virtio interface to the guest kernel for accessing the network interface, instead of simulating a real network interface card. This can improve network I/O performance, but it requires a Virtio driver in guest kernel. Use it instead of a plain `-net` if the guest kernel supports Virtio.

Controlling the Emulated Display

To set the resolution of the emulated display for a Xorg server running in the guest Linux system, read [the section called "Fine Tuning Display Settings"](#).

Networking

The above solution for networking allows the guest to access the local network through the host (and possibly to access internet through the local routers), but the converse is not true. Not even the host can access the guest, unless port forwarding is enabled. And in the case several guests are running, they cannot communicate with each other. Other network devices can be used for this purpose. For example, there is the "socket" device, which allows several guests to share a common virtual network. In the following, we describe in more details how to set up the "bridge" device, which allows the guests to appear as if connected to the local network. All the commands below should be run as the `root` user.

Allow the host to forward IP packets:

```
sysctl -w net.ipv4.ip_forward=1
```

To make this permanent, add the command to `/etc/sysctl.d/60-net-forward.conf`:

```
cat >> /etc/sysctl.d/60-net-forward.conf << EOF
net.ipv4.ip_forward=1
EOF
```

Set up a required configuration file:

```
install -vdm 755 /etc/qemu &&
echo allow br0 > /etc/qemu/bridge.conf
```

In the qemu command line above, replace the switch `-netdev user,...` with `-netdev bridge,...`

Contents

Installed Programs: elf2dmp, qemu (symlink), qemu-edid, qemu-ga, qemu-img, qemu-io, qemu-keymap, qemu-nbd, qemu-pr-helper, qemu-storage-daemon, and qemu-system-<arch>

Installed Library: None

Installed Directories: /usr/share/qemu and /usr/share/doc/qemu-9.0.2 (optional)

Short Description

<code>elf2dmp</code>	Converts files from elf to dmp format
<code>qemu-edid</code>	is a test tool for the qemu EDID generator
<code>qemu-ga</code>	implements support for QMP (QEMU Monitor Protocol) commands and events that terminate and originate respectively within the guest using an agent built as part of QEMU
<code>qemu-img</code>	provides commands to manage QEMU disk images
<code>qemu-io</code>	is a diagnostic and manipulation program for (virtual) memory media. It is still at an early stage of development
<code>qemu-keymap</code>	generates qemu reverse keymaps from xkb keymaps, which can be used with the qemu "-k" command line switch
<code>qemu-nbd</code>	exports Qemu disk images using the QEMU Disk Network Block Device (NBD) protocol
<code>qemu-pr-helper</code>	Implements the persistent reservation helper for QEMU
<code>qemu-storage-daemon</code>	allows to modify disk images using the QEMU Monitor Protocol (QMP) without running a VM

```
qemu-system-  
x86_64      is the QEMU PC System emulator
```

Part III. General Libraries and Utilities

Chapter 9. General Libraries

Libraries contain code which is often required by more than one program. This has the advantage that each program doesn't need to duplicate code (and risk introducing bugs), it just has to call functions from the libraries installed on the system. The most obvious example of a set of libraries is Glibc which is installed during the LFS book. This contains all of the C library functions which programs use.

There are two types of libraries: static and shared. Shared libraries (usually `libXXX.so`) are loaded into memory from the shared copy at runtime (hence the name). Static libraries (`libXXX.a`) are actually linked into the program executable file itself, thus making the program file larger. Quite often, you will find both static and shared copies of the same library on your system.

Generally, you only need to install libraries when you are installing software that needs the functionality they supply. In the BLFS book, each package is presented with a list of (known) dependencies. Thus, you can figure out which libraries you need to have before installing that program. If you are installing something without using BLFS instructions, usually the `README` or `INSTALL` file will contain details of the program's requirements.

There are certain libraries which nearly *everyone* will need at some point. In this chapter these and some others are listed and it is explained why you may want to install them.

Abseil-cpp-20240722.0

Introduction to Abseil-cpp

The Abseil-cpp package contains a series of libraries designed to augment the C++ standard library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/abseil/abseil-cpp/releases/download/20240722.0/abseil-cpp-20240722.0.tar.gz>
- Download MD5 sum: 740fb8f35ebdf82740c294bde408b9c0
- Download size: 2.1 MB
- Estimated disk space required: 32 MB
- Estimated build time: 0.9 SBU

Abseil-cpp Dependencies

Required

[CMake-3.30.2](#)

Installation of Abseil-cpp

Install Abseil-cpp by running the following commands:

```
mkdir build &&  
cd      build &&  
  
cmake -D CMAKE_INSTALL_PREFIX=/usr \  
      -D CMAKE_BUILD_TYPE=Release \  
      -D ABSL_PROPAGATE_CXX_STD=ON \  
      -D BUILD_SHARED_LIBS=ON \  
      -G Ninja ...           &&  
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D ABSL_PROPAGATE_CXX_STD=ON`: This parameter enables propagating C++ features to targets that link to this package's libraries.

`-D BUILD_SHARED_LIBS=ON`: This parameter builds shared versions of the libraries provided by this package instead of static libraries.

Contents

Installed Programs: None

Installed Libraries: libabsl_bad_any_cast_impl.so, libabsl_bad_optional_access.so, libabsl_bad_variant_access.so, libabsl_base.so, libabsl_city.so, libabsl_civil_time.so, libabsl_cord_internal.so, libabsl_cord.so, libabsl_cordz_functions.so, libabsl_cordz_handle.so, libabsl_cordz_info.so, libabsl_cordz_sample_token.so, libabsl_crc32c.so, libabsl_crc_cord_state.so, libabsl_crc_cpu_detect.so, libabsl_crc_internal.so, libabsl_debugging_internal.so, libabsl_decode_rust_punycode.so, libabsl_demangle_internal.so, libabsl_demangle_rust.so, libabsl_die_if_null.so, libabsl_examine_stack.so, libabsl_exponential_biased.so, libabsl_failure_signal_handler.so, libabsl_flags_commandlineflag_internal.so, libabsl_flags_commandlineflag.so, libabsl_flags_config.so, libabsl_flags_internal.so, libabsl_flags_marshall.so, libabsl_flags_parse.so, libabsl_flags_private_handle_accessor.so, libabsl_flags_program_name.so, libabsl_flags_reflection.so, libabsl_flags_usage_internal.so, libabsl_flags_usage.so, libabsl_graphcycles_internal.so, libabsl_hash.so, libabsl_hashtablez_sampler.so, libabsl_int128.so, libabsl_kernel_timeout_internal.so, libabsl_leak_check.so, libabsl_log_entry.so, libabsl_log_flags.so, libabsl_log_globals.so, libabsl_log_initialize.so, libabsl_log_internal_check_op.so, libabsl_log_internal_conditions.so, libabsl_log_internal_fnmatch.so, libabsl_log_internal_format.so, libabsl_log_internal_globals.so, libabsl_log_internal_log_sink_set.so, libabsl_log_internal_message.so, libabsl_log_internal_nullguard.so, libabsl_log_internal_proto.so, libabsl_log_severity.so, libabsl_log_sink.so, libabsl_low_level_hash.so, libabsl_malloc_internal.so, libabsl_periodic_sampler.so, libabsl_poison.so, libabsl_random_distributions.so, libabsl_random_internal_distribution_test_util.so, libabsl_random_internal_platform.so, libabsl_random_internal_pool_urbg.so, libabsl_random_internal_randen_hwaes_impl.so, libabsl_random_internal_randen_hwaes.so, libabsl_random_internal_randen_slow.so, libabsl_random_internal_randen.so, libabsl_random_internal_seed_material.so, libabsl_random_seed_gen_exception.so, libabsl_random_seed_sequences.so, libabsl_raw_hash_set.so, libabsl_raw_logging_internal.so, libabsl_scoped_set_env.so, libabsl_spinlock_wait.so, libabsl_stacktrace.so, libabsl_statusor.so, libabsl_status.so, libabsl_strerror.so, libabsl_str_format_internal.so, libabsl_strings_internal.so, libabsl_strings.so, libabsl_string_view.so, libabsl_symbolize.so, libabsl_synchronization.so, libabsl_throw_delegate.so, libabsl_time.so, libabsl_time_zone.so, libabsl_utf8_for_code_point.so, and libabsl_vlog_config_internal.so

Installed Directories: /usr/include/absl, /usr/lib/cmake/absl

AppStream-1.0.3

Introduction to AppStream

The AppStream package contains a library and tool that is useful for retrieving software metadata and making it easily accessible to programs which need it.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/appstream/releases/AppStream-1.0.3.tar.xz>
- Download MD5 sum: 8db15579a7b25acca99677ddc6a12b68
- Download size: 2.5 MB
- Estimated disk space required: 30 MB
- Estimated build time: 0.5 SBU (With tests; both using parallelism=4)

AppStream Dependencies

Required

[cURL-8.9.1](#), [ltsstool-2.0.7](#), [libxml2-2.13.3](#), [libxmlb-0.3.19](#), and [libyaml-0.2.5](#)

Optional

[Gi-DocGen-2024.1](#), [Qt-6.7.2](#), [DAPS](#), and [libstemmer](#)

Installation of AppStream

Install AppStream by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D apidocs=false    \
            -D stemming=false   ... &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install &&
mv -v /usr/share/doc/appstream{,-1.0.3}
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D apidocs=false`: This switch disables building the API documentation. Remove it if you have [Gi-DocGen-2024.1](#) installed and wish to regenerate the API documentation. When the API documentation is not regenerated, a pre-built copy is installed anyway.

`-D stemming=false`: This switch disables stemming support. Remove this switch if you have [libstemmer](#) installed and want faster searches.

`-D qt5=true`: Use this option if you have [qt5-components-5.15.14](#) installed and you want to build support for Qt-5 applications into this package.

`-D qt=true`: Use this option if you have [Qt-6.7.2](#) installed and you want to build support for Qt6 applications into this package. This option conflicts with `-D qt5=true`.

Configuring AppStream

Config Files

`/usr/share/metainfo/org.linuxfromscratch.lfs.xml`

Configuration Information

AppStream expects an operating system metainfo file describing the GNU/Linux distribution. As the `root` user, create the file describing LFS:

```
install -vdm755 /usr/share/metainfo &&
cat > /usr/share/metainfo/org.linuxfromscratch.lfs.xml << EOF
<?xml version="1.0" encoding="UTF-8"?>
<component type="operating-system">
  <id>org.linuxfromscratch.lfs</id>
  <name>Linux From Scratch</name>
  <summary>A customized Linux system built entirely from source</summary>
  <description>
    <p>
      Linux From Scratch (LFS) is a project that provides you with
      step-by-step instructions for building your own customized Linux
      system entirely from source.
    </p>
  </description>
  <url type="homepage">https://www.linuxfromscratch.org/lfs/</url>
  <metadata_license>MIT</metadata_license>
  <developer id='linuxfromscratch.org'>
    <name>The Linux From Scratch Editors</name>
  </developer>
</component>
```

```

</developer>

<releases>
  <release version="12.2" type="release" date="2024-04-01">
    <description>
      <p>Now contains Binutils 2.43.1, GCC-14.2.0, Glibc-2.40,
      and Linux kernel 6.10.</p>
    </description>
  </release>

  <release version="12.1" type="stable" date="2024-03-01">
    <description>
      <p>Now contains Binutils 2.42, GCC-13.2.0, Glibc-2.39,
      and Linux kernel 6.7.</p>
    </description>
  </release>
</releases>
</component>
EOF

```

Contents

Installed Programs: appstreamcli

Installed Libraries: libappstream.so

Installed Directories: /usr/include/appstream, /usr/share/doc/appstream-1.0.3, and /usr/share/installed-tests/appstream

Short Descriptions

appstreamcli libappstream.so	queries information from AppStream metadata and from the AppStream component index contains functions that handle AppStream metadata queries and request information from the AppStream component index
---	--

appstream-glib-0.8.3

Introduction to appstream-glib

The appstream-glib provides GObjects and helper methods to make it easy to read and write AppStream metadata. It also provides a simple DOM (Document Object Model) implementation that makes it easy to edit nodes and convert to and from the standardized XML representation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://people.freedesktop.org/~hughsient/appstream-glib/releases/appstream-glib-0.8.3.tar.xz>
- Download MD5 sum: 2ffd46eff1c16f31e435849b706c2287
- Download size: 2.2 MB
- Estimated disk space required: 15 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

appstream-glib Dependencies

Required

[cURL-8.9.1](#), [gdk-pixbuf-2.42.12](#), and [libarchive-3.7.4](#)

Optional

[docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [GTK-Doc-1.34.0](#), [libxslt-1.1.42](#), and [libyaml-0.2.5](#)

Installation of appstream-glib

Install appstream-glib by running the following commands:

```

mkdir build &&
cd      build &&

```

```
meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D rpm=false \
    && \
    ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install &&
rm -v -rf /usr/share/installed-tests
```

Contents

Installed Programs: appstream-builder, appstream-compose, and appstream-util

Installed Libraries: libappstream-glib.so

Installed Directories: /usr/include/libappstream-glib and /usr/lib/asb-plugins-5

Short Descriptions

<code>appstream-builder</code>	builds AppStream metadata
<code>appdata-compose</code>	generates AppStream metadata
<code>appstream-util</code>	Is a management tool for <code>appstream-builder</code>
<code>libappstream-glib.so</code>	contains the API functions

Apr-1.7.4

Introduction to Apr

The Apache Portable Runtime (APR) is a supporting library for the Apache web server. It provides a set of application programming interfaces (APIs) that map to the underlying Operating System (OS). Where the OS doesn't support a particular function, APR will provide an emulation. Thus programmers can use the APR to make a program portable across different platforms.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist/apr/apr-1.7.4.tar.bz2>
- Download MD5 sum: f8a62f3984898ba0ea8b6f26b851cb99
- Download size: 876 KB
- Estimated disk space required: 11 MB (additional 4 MB for the tests)
- Estimated build time: 0.1 SBU (add 1.4 SBU for tests)

Installation of Apr

Install Apr by running the following commands:

```
./configure --prefix=/usr \
    --disable-static \
    --with-installbuilddir=/usr/share/apr-1/build &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: apr-1-config

Installed Library: libapr-1.so

Installed Directories: /usr/include/apr-1 and /usr/share/apr-1

Short Descriptions

apr-1- config	is a shell script used to retrieve information about the apr library in the system. It is typically used to compile and link against the library
libapr-1.so	is the Apache Portable Runtime library

Apr-Util-1.6.3

Introduction to Apr Util

The Apache Portable Runtime Utility Library provides a predictable and consistent interface to underlying client library interfaces. This application programming interface assures predictable if not identical behavior regardless of which libraries are available on a given platform.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist/apr/apr-util-1.6.3.tar.bz2>
- Download MD5 sum: b6e8c9b31d938fe5797ceb0d1ff2eb69
- Download size: 423 KB
- Estimated disk space required: 7.6 MB (add 1.4 MB for tests)
- Estimated build time: less than 0.1 SBU (add 0.3 SBU for tests)

Apr Util Dependencies

Required

[Apr-1.7.4](#)

Optional

[FreeTDS](#), [MariaDB-10.11.8](#) or [MySQL](#), [OpenLDAP-2.6.8](#), [PostgreSQL-16.4](#), [SQLite-3.46.1](#), [unixODBC-2.3.12](#), and [Berkeley DB](#) (deprecated)

Installation of Apr Util

Install Apr Util by running the following commands:

```
./configure --prefix=/usr      \
            --with-apr=/usr    \
            --with-gdbm=/usr   \
            --with-openssl=/usr \
            --with-crypto &&
make
```

To test the results, issue: `make -j1 test`. One test, `testdbm`, is known to fail.

Now, as the `root` user:

```
make install
```

Command Explanations

`--with-gdbm=/usr`: This switch enables the `apr_dbm_gdbm-1.so` plugin.

`--with-openssl=/usr --with-crypto`: These switches enable the `apr_crypto_openssl-1.so` plugin.

`--with-berkeley-db=/usr`: If you have installed [Berkeley DB](#) (deprecated), use this switch to compile the `apr_dbm_db-1.so` plugin.

--with-ldap: If you have installed [OpenLDAP-2.6.8](#), use this switch to compile the `apr_ldap.so` plugin.

Contents

Installed Program: `apu-1-config`

Installed Library: `libaprutil-1.so`

Installed Directory: `/usr/lib/apr-util-1`

Short Descriptions

`apu-1-config` is an APR-util script designed to allow easy command line access to APR-util configuration parameters

`libaprutil-1.so` contains functions that provide a predictable and consistent interface to underlying client library interfaces

Aspell-0.60.8.1

Introduction to Aspell

The Aspell package contains an interactive spell checking program and the Aspell libraries. Aspell can either be used as a library or as an independent spell checker.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/aspell/aspell-0.60.8.1.tar.gz>
- Download MD5 sum: 187bd142f522ada555c7aa6b9cbf56e6
- Download size: 3.4 MB
- Estimated disk space required: 41 MB (Additional 8 MB for EN dictionary)
- Estimated build time: 0.4 SBU

Additional Downloads

You'll need to download at least one dictionary. The English dictionary is given as an example below. Dictionaries in many other languages can be found at <https://ftp.gnu.org/gnu/aspell/dict>.

- Aspell English dictionary: <https://ftp.gnu.org/gnu/aspell/dict/en/aspell6-en-2020.12.07-0.tar.bz2>

Aspell Dependencies

Required

[Which-2.21](#) (for the dictionaries)

Installation of Aspell

Install Aspell by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
ln -svfn aspell-0.60 /usr/lib/aspell &&

install -v -m755 -d /usr/share/doc/aspell-0.60.8.1/aspell{,-dev}.html &&

install -v -m644 manual/aspell.html/* \
/usr/share/doc/aspell-0.60.8.1/aspell.html &&

install -v -m644 manual/aspell-dev.html/* \
/usr/share/doc/aspell-0.60.8.1/aspell-dev.html
```

If you do not plan to install Ispell, then copy the wrapper script `ispell`:

```
install -v -m 755 scripts/ispell /usr/bin/
```

If you do not plan to install Spell, then copy the wrapper script `spell`:

```
install -v -m 755 scripts/spell /usr/bin/
```

Command Explanations

`ln -svfn aspell-0.60 /usr/lib/aspell`: This command is useful for configuration of other applications, such as [enchant-2.8.2](#).

Configuring Aspell

Configuration Information

After Aspell is installed, you must set up at least one dictionary. Install the English dictionary by running the following commands:

```
tar xf ../aspell16-en-2020.12.07-0.tar.bz2 &&
cd aspell16-en-2020.12.07-0 &&
./configure &&
make
```

Now, as the `root` user:

```
make install
```

Other dictionaries can be installed with the same instructions.

Contents

Installed Programs: aspell, aspell-import, precat, preunzip, prezzip, prezzip-bin, pspell-config, run-with-aspell, word-list-compress and optionally, ispell and spell.

Installed Libraries: libaspell.so and libpspell.so

Installed Directories: /usr/include/pspell and /usr/lib/aspell-0.60

Short Descriptions

<code>aspell</code>	is a utility that can function as an <code>ispell -a</code> replacement, as an independent spell checker, as a test utility to test out Aspell features, and as a utility for managing dictionaries
<code>ispell</code>	is a wrapper around <code>aspell</code> to invoke it in <code>ispell</code> compatible mode
<code>spell</code>	is a wrapper around <code>aspell</code> to invoke it in <code>spell</code> compatible mode
<code>aspell-import</code>	imports old personal dictionaries into Aspell
<code>precat</code>	decompresses a <code>prezip</code> ped file to stdout
<code>preunzip</code>	decompresses a <code>prezip</code> ped file
<code>prezip</code>	is a prefix delta compressor, used to compress sorted word lists or other similar text files
<code>prezip-bin</code>	is called by the various wrapper scripts to perform the actual compressing and decompressing
<code>pspell-config</code>	displays information about the <code>libpspell</code> installation, mostly for use in build scripts
<code>run-with-aspell</code>	is a script to help use Aspell as an <code>ispell</code> replacement
<code>word-list-compress</code>	compresses or decompresses sorted word lists for use with the Aspell spell checker
<code>libaspell.so</code>	contains spell checking API functions
<code>libpspell.so</code>	is an interface to the <code>libaspell</code> library. All the spell checking functionality is now in <code>libaspell</code> but this library is included for backward compatibility

boost-1.86.0

Introduction to Boost

Boost provides a set of free peer-reviewed portable C++ source libraries. It includes libraries for linear algebra, pseudorandom number generation, multithreading, image processing, regular expressions and unit testing.

This package is known to build and work properly using an LFS 12.2 platform.

Note

This package will extract to the `boost-1.86.0/` directory.

Package Information

- Download (HTTP): <https://github.com/boostorg/boost/releases/download/boost-1.86.0/boost-1.86.0-b2-nodocs.tar.xz>
- Download MD5 sum: 7ad40a9d9d1db59ee2ed62e8fa7134ed
- Download size: 45 MB
- Estimated disk space required: 899 MB (197 MB installed)
- Estimated build time: 1.8 SBU (Using parallelism=4; add 0.1 SBU for tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/boost-1.86.0-upstream_fixes-1.patch

Boost Dependencies

Recommended

[Which-2.21](#)

Optional

[ICU-75.1](#), [NumPy-2.1.0](#), and [Open MPI](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/boost>

Installation of Boost

First, fix a problem with Boost and [NumPy-2.1.0](#) that causes the build to fail:

```
patch -Np1 -i ../boost-1.86.0-upstream_fixes-1.patch
```

Next, fix a build issue which occurs in the stacktrace library. This issue is specific to i686 systems.

```
case $(uname -m) in
  i?86)
    sed -e "s/defined(__MINGW32__)/& || defined(__i386__)/" \
          -i ./libs/stacktrace/src/exception_headers.h ;;
esac
```

This package can be built with several jobs running in parallel. In the instructions below, `<N>` stands for the number of jobs. Install Boost by running the following commands:

```
./bootstrap.sh --prefix=/usr --with-python=python3 &&
./b2 stage -j<N> threading=multi link=shared
```

To run Boost.Build's regression tests, issue `pushd tools/build/test; python3 test_all.py; popd`.

To run every library's regression tests, issue `pushd status; ..b2; popd`. A few tests may fail. They take a very long time (over 119 SBU at -j4) and use a very large amount of disk space (46 GB). You should use the `-jN` switch to speed them up.

Note

Boost installs many versioned directories in /usr/lib/cmake. If a new version of Boost is installed over a previous version, the older cmake directories need to be explicitly removed. To do this, run as the `root` user:

```
rm -rf /usr/lib/cmake/[Bb]oost*
```

before installing the new version.

Now, as the `root` user:

```
./b2 install threading=multi link=shared
```

Command Explanations

`threading=multi`: This parameter ensures that Boost is built with multithreading support.

`link=shared`: This parameter ensures that only shared libraries are created, except for `libboost_exception` and `libboost_test_exec_monitor` which are created as static. Most people will not need the static libraries, and most programs using Boost only use the headers. Omit this parameter if you do need static libraries.

`--with-python=python3`: This switch ensures Python3 is used if Python2 is installed.

`-jN`: This switch may be added to the `b2` command lines, to run up to N processes in parallel.

Contents

Installed Programs: None

Installed Libraries: `libboost_atomic.so`, `libboost_charconv.so`, `libboost_chrono.so`, `libboost_container.so`, `libboost_context.so`, `libboost_contract.so`, `libboost_coroutine.so`, `libboost_date_time.so`, `libboost_exception.a`, `libboost_fiber.so`, `libboost_filesystem.so`, `libboost_graph.so`, `libboost_iostreams.so`, `libboost_json.so`, `libboost_locale.so`, `libboost_log_setup.so`, `libboost_log.so`, `libboost_math_c99.so`, `libboost_math_c99f.so`, `libboost_math_c99l.so`, `libboost_math_tr1.so`, `libboost_math_tr1f.so`, `libboost_math_tr1l.so`, `libboost_nowide.so`, `libboost_numpy312.so`, `libboost_prg_exec_monitor.so`, `libboost_program_options.so`, `libboost_python312.so`, `libboost_random.so`, `libboost_regex.so`, `libboost_serialization.so`, `libboost_stacktrace_addr2line.so`, `libboost_stacktrace_basic.so`, `libboost_stacktrace_noop.so`, `libboost_system.so`, `libboost_test_exec_monitor.a`, `libboost_thread.so`, `libboost_timer.so`, `libboost_type_erasure.so`, `libboost_unit_test_framework.so`, `libboost_url.so`, `libboost_wave.so`, and `libboost_wserialization.so`

Installed Directory: `/usr/include/boost`

brotli-1.1.0

Introduction to Brotli

Brotli provides a general-purpose lossless compression algorithm that compresses data using a combination of a modern variant of the LZ77 algorithm, Huffman coding and 2nd order context modeling. Its libraries are particularly used for WOFF2 fonts on webpages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/google/brotli/archive/v1.1.0/brotli-1.1.0.tar.gz>
- Download MD5 sum: 3a6a3dba82a3604792d3cb0bd41bca60
- Download size: 500 KB
- Estimated disk space required: 33 MB (with python3 bindings)
- Estimated build time: 0.3 SBU (with python3 bindings; parallelism=4)

Brotli Dependencies

Required

[CMake-3.30.2](#)

Optional

[pytest-8.3.2](#) (for testing Python3 bindings)

Installation of Brotli

Install brotli by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      .. &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

If desired, build the Python3 bindings:

```
cd .. &&
sed "/c\/*.[ch]'/d; \
/include_dirs=\\[\\ \
i libraries=['brotlicommon','brotlidec','brotlienc'],\" \
-i setup.py &&
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Install the Python3 bindings as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user Brotli
```

To test the Python3 binding, issue: `pytest`.

Command Explanations

`sed ... -i setup.py`: Stop `setup.py` from rebuilding the entire package all over again, use the already installed libraries for the Python3 binding instead.

Contents

Installed Programs: brotli

Installed Libraries: libbrotlicommon.so, libbrotlidec.so, and libbrotlienc.so

Installed Directories: /usr/include/brotli and /usr/lib/python3.12/site-packages/Brotli-1.1.0.dist-info (if you built and installed the Python3 bindings)

Short Descriptions

<code>brotli</code>	can compress or decompress files, or test the integrity of compressed files
<code>libbrotlicommon(-static.a,.so)</code>	is the Brotli common dictionary library
<code>libbrotlidec(-static.a,.so)</code>	is the Brotli decoder library
<code>libbrotlienc(-static.a,.so)</code>	is the Brotli common encoder library

CLucene-2.3.3.4

Introduction to CLucene

CLucene is a C++ version of Lucene, a high performance text search engine.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/clucene/clucene-core-2.3.3.4.tar.gz>
- Download MD5 sum: 48d647fb8ef8889e5a7f422c1bfda94
- Download size: 2.2 MB
- Estimated disk space required: 78 MB

- Estimated build time: 0.8 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/clucene-2.3.3.4-contribs_lib-1.patch

CLucene Dependencies

Required

[CMake-3.30.2](#)

Recommended

[Boost-1.86.0](#)

Installation of CLucene

Install CLucene by running the following commands:

```
patch -Np1 -i ../clucene-2.3.3.4-contribs_lib-1.patch &&
sed -i '/Misc.h/a #include <ctime>' src/core/CLucene/document/DateTools.cpp &&
mkdir build &&
cd build &&
cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D BUILD_CONTRIBS_LIB=ON .. &&
make
```

Now, as the `root` user:

```
make install
```

Command Explanations

`-D BUILD_CONTRIBS_LIB=ON`: This cmake variable enables building the CLucene contribs library necessary for running applications that use language specific text analyzers like LibreOffice for example.

Contents

Installed Programs: None

Installed Libraries: libclucene-contribs-lib.so, libclucene-core.so, and libclucene-shared.so

Installed Directories: /usr/include/CLucene and /usr/lib/CLuceneConfig.cmake

dbus-glib-0.112

Introduction to D-Bus GLib

The D-Bus GLib package contains GLib interfaces to the D-Bus API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://dbus.freedesktop.org/releases/dbus-glib/dbus-glib-0.112.tar.gz>
- Download MD5 sum: 021e6c8a288df02c227e4aafbf7e7527
- Download size: 700 KB
- Estimated disk space required: 12 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

D-Bus GLib Dependencies

Required

[dbus-1.14.10](#) and [GLib-2.80.4](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of D-Bus GLib

Install D-Bus GLib by running the following commands:

```
./configure --prefix=/usr \
--sysconfdir=/etc \
--disable-static &&
make
```

To test the results, issue: `make check`. Note that more comprehensive tests can be run by following the same method used in D-Bus instructions, which requires building the package twice.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Program: `dbus-binding-tool`

Installed Library: `libdbus-glib-1.so`

Installed Directories: `/usr/share/gtk-doc/html/dbus-glib`

Short Descriptions

<code>dbus-binding-tool</code>	is a tool used to interface with the D-Bus API
<code>libdbus-glib-1.so</code>	contains GLib interface functions to the D-Bus API

Double-conversion-3.3.0

Introduction to Double-conversion

The Double-conversion package contains a library that facilitates binary-to-decimal and decimal-to-binary routines for IEEE doubles.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/google/double-conversion/archive/v3.3.0/double-conversion-3.3.0.tar.gz>
- Download MD5 sum: b344abb64084a4a1d98a43e67752989b
- Download size: 6.7 MB
- Estimated disk space required: 62 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Double-conversion Dependencies

Required

[CMake-3.30.2](#)

Installation of Double-conversion

Install Double-conversion by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D BUILD_SHARED_LIBS=ON \
-D BUILD_TESTING=ON \
..
&&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D BUILD_SHARED_LIBS=ON`: This switch forces cmake to build a shared version of the library instead of the static version.

`-D BUILD_TESTING=ON`: This switch builds the test programs.

Contents

Installed Programs: None

Installed Libraries: libdouble-conversion.so

Installed Directories: /usr/include/double-conversion

Short Descriptions

`libdouble-conversion.so` provides binary-to-decimal and decimal-to-binary routines for IEEE doubles

duktape-2.7.0

Introduction to duktape

duktape is an embeddable Javascript engine, with a focus on portability and compact footprint.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://duktape.org/duktape-2.7.0.tar.xz>
- Download MD5 sum: b3200b02ab80125b694bae887d7c1ca6
- Download size: 1003 KB
- Estimated disk space required: 25 MB
- Estimated build time: 0.3 SBU

Installation of duktape

Install duktape by running the following commands:

```
sed -i 's/-Os/-O2/' Makefile.sharedlibrary
make -f Makefile.sharedlibrary INSTALL_PREFIX=/usr
```

Now, as the `root` user:

```
make -f Makefile.sharedlibrary INSTALL_PREFIX=/usr install
```

Contents

Installed Programs: None

Installed Libraries: libduktape.so and libduktape.so

Installed Directories: None

Short Descriptions

`libduktape.so` is an embeddable Javascript engine

enchant-2.8.2

Introduction to enchant

The enchant package provides a generic interface into various existing spell checking libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/AbiWord/enchant/releases/download/v2.8.2/enchant-2.8.2.tar.gz>
 - Download MD5 sum: 92dcfe06feb92a3d4bbff4e08b08d3d
 - Download size: 1.3 MB
 - Estimated disk space required: 9.4 MB (add 77 MB for tests)
 - Estimated build time: 0.1 SBU (add 0.9 SBU for tests)

enchant Dependencies

Required

GLib-2.80.4

Recommended

Aspell-0.60.8.1

Optional

[dbus-glib-0.112](#), [Doxygen-1.12.0](#), [Hspell](#), [Hunspell](#), [Nuspell](#), [Voikko](#), and [unittest-cpp](#) (required for tests)

Installation of enchant

Install enchant by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --docdir=/usr/share/doc/enchant-2.8.2 &&
make
```

To run tests, `unittest-cpp` must be installed and the `--enable-relocatable` option passed to `configure` above. If these conditions are present, the tests may be run with `make check`.

Now, as the *root* user:

`make install`

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Configuring enchant

Config Files

```
~/.enchant and /usr/share/enchant/enchant.ordering
```

Configuration Information

You can test your installation and configuration by creating a test file and running the following commands (you can replace the en_GB dictionary by any other downloaded when installing [Aspell-0.60.8.1](#)):

```
cat > /tmp/test-enchant.txt << "EOF"
Tel me more abot linux
Ther ar so many commads
EOF

enchant-2 -d en_GB -l /tmp/test-enchant.txt &&
enchant-2 -d en_GB -a /tmp/test-enchant.txt
```

You will see a list of the misspelled words followed by a list of alternatives for them.

See more details in the enchant manual page.

Contents

Installed Programs: enchant-2 and enchant-lsmod-2

Installed Libraries: libenchant-2.so

Installed Directories: /usr/include/enchant-2, /usr/lib/enchant-2, /usr/share/enchant, and /usr/share/doc/enchant-2.8.2

Short Descriptions

enchant-2	is a spellchecker
enchant-lsmod-2	lists available backends, languages, and dictionaries
libenchant-2.so	contains spell checking interface API functions

Exempi-2.6.5

Introduction to Exempi

Exempi is an implementation of XMP (Adobe's Extensible Metadata Platform).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://libopenraw.freedesktop.org/download/exempi-2.6.5.tar.xz>
- Download MD5 sum: 51fe14c2a5fa44816ba8187c6ad87d78
- Download size: 2.7 MB
- Estimated disk space required: 289 MB (add 236 MB for tests)
- Estimated build time: 0.4 SBU (add 0.6 SBU for tests; both using parallelism=4)

Exempi Dependencies

Required

[Boost-1.86.0](#)

Optional

[Valgrind-3.23.0](#)

Installation of Exempi

If you intend to run the regression tests, first remove a test that depends on a proprietary Adobe SDK:

```
sed -i -r '/^\\s?testadobesdk/d' exempi/Makefile.am &&
autoreconf -fiv
```

Install Exempi by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: `exempi`

Installed Library: `libexempi.so`

Installed Directory: `/usr/include/exempi-2.0`

Short Descriptions

`exempi` is a command line tool to manipulate XMP metadata
`libexempi.so` is a library used to parse XMP metadata

fftw-3.3.10

Introduction to fftw

FFTW is a C subroutine library for computing the discrete Fourier transform (DFT) in one or more dimensions, of arbitrary input size, and of both real and complex data (as well as of even/odd data, i.e. the discrete cosine/sine transforms or DCT/DST).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.fftw.org/fftw-3.3.10.tar.gz>
- Download MD5 sum: 8ccbf6a5ea78a16dbc3e1306e234cc5c
- Download size: 4.0 MB
- Estimated disk space required: 59 MB
- Estimated build time: 1.6 SBU (add 3.4 SBU for tests, both using parallelism=4)

Installation of fftw

Note

We build fftw three times for different libraries in different numerical precisions: the default double precision floating point, the older 32-bit (single precision) version named float which sacrifices precision for speed, and the long double which offers increased precision at the cost of slower execution.

The first build is for double precision arithmetic. Install fftw by running the following commands:

```
./configure --prefix=/usr      \
--enable-shared   \
--disable-static \
--enable-threads \
--enable-sse2     \
--enable-avx      \
--enable-avx2     &&
make
```

To test the results, issue: `make check`. On 32-bit systems, the tests can take substantially longer than they would on 64-bit machines.

Now, as the `root` user:

```
make install
```

Now build single precision:

```
make clean &&  
  
.configure --prefix=/usr    \  
    --enable-shared    \  
    --disable-static    \  
    --enable-threads    \  
    --enable-sse2    \  
    --enable-avx    \  
    --enable-avx2    \  
    --enable-float    &&  
  
make
```

As the `root` user:

```
make install
```

Finally, build long double precision:

```
make clean &&  
  
.configure --prefix=/usr    \  
    --enable-shared    \  
    --disable-static    \  
    --enable-threads    \  
    --enable-long-double &&  
  
make
```

As the `root` user:

```
make install
```

Command Explanations

`--enable-shared --disable-static`: Use shared libs instead of static libs.

`--enable-threads`: This enables `libfftw3_threads.so` to be compiled. It is used by e.g. the gimp plugin from [G'MIC](#).

`--enable-{sse2,avx,avx2}`: These enables building the optimized routines using SSE2, AVX, and AVX2 instructions. FFTW will check if these routines can be really used on the current CPU when the FFTW library is loaded, so a FFTW build with these routines enabled can still run on a CPU without SSE2, AVX, or AVX2. These options are not compatible with `--enable-long-double`.

`--enable-float`: This enables building the library that uses single precision floating point arithmetic. It is faster but less precise than the default double precision library. The library will be called `libfftw3f.so` needed by [PulseAudio-17.0](#).

`--enable-long-double`: This enables building the library that uses higher precision long-double floating point arithmetic. The library will be called `libfftw3l.so`.

`--enable-avx512`: This enables building the optimized routines using AVX512F instructions. FFTW will check if these routines can be really used on the current CPU when the FFTW library is loaded, so a FFTW build with these routines enabled can still run on a CPU without AVX512F. Use this option if the FFTW build will be used on a CPU with AVX512F. This option is not compatible with `--enable-long-double`.

Contents

Installed Programs: `fftw-wisdom` and `fftw-wisdom-to-conf`

Installed Libraries: `libfftw3.so`, `libfftw3_threads.so`, `libfftw3f.so`, `libfftw3f_threads.so`, `libfftw3l.so` and `libfftw3l_threads.so`

Installed Directories: None

Short Descriptions

<code>fftw-wisdom</code>	is a utility to generate FFTW wisdom files, which contain saved information about how to optimally compute (Fourier) transforms of various sizes
<code>fftw-wisdom-to-conf</code>	is a utility to generate C configuration routines from FFTW wisdom files, where the latter contain saved information about how to optimally compute (Fourier) transforms of various sizes
<code>libfftw3.so</code>	is the Fast Fourier Transform library
<code>libfftw3_threads.so</code>	is the threaded Fast Fourier Transform library
<code>libfftw3f.so</code>	is the single-precision Fast Fourier Transform library, described as "float" for historic reasons
<code>libfftw3f_threads.so</code>	is the threaded single-precision Fast Fourier Transform library
<code>libfftw3l.so</code>	is the long double Fast Fourier Transform library
<code>libfftw3l_threads.so</code>	is the threaded long double Fast Fourier Transform library

fmt-11.0.2

Introduction to fmt

The fmt package is an open-source formatting library providing a fast and safe alternative to C stdio and C++ iostreams.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/fmtlib/fmt/archive/11.0.2/fmt-11.0.2.tar.gz>
- Download MD5 sum: 3fe10c5184c8ecd0d2f9536c1b1ae95c
- Download size: 688 KB
- Estimated disk space required: 3.9 MB (add 39 MB for tests)
- Estimated build time: less than 0.1 SBU (add 0.2 SBU for tests; both using parallelism=4)

Installation of fmt

Install fmt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
      -D CMAKE_INSTALL_LIBDIR=/usr/lib \
      -D BUILD_SHARED_LIBS=ON          \
      -D FMT_TEST=OFF                \
      -G Ninja ...                  &&
ninja
```

If you have enabled tests, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D FMT_TEST=OFF`: This switch initializes the package tests. Set to ON if you wish to run tests.

Contents

Installed Programs: None

Installed Libraries: `libfmt.so`

Installed Directories: /usr/include/fmt and /usr/lib/cmake/fmt

GLib-2.80.4

Introduction to GLib

The GLib package contains low-level libraries useful for providing data structure handling for C, portability wrappers and interfaces for runtime functionality such as an event loop, threads, dynamic loading and an object system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/glib/2.80/glib-2.80.4.tar.xz>
- Download MD5 sum: 4334211338220a165350d1c4a1597b0e
- Download size: 5.3 MB
- Estimated disk space required: 189 MB (add 22 MB for tests)
- Estimated build time: 0.9 SBU (add 0.4 SBU for tests; both using parallelism=4)

Additional Downloads

GObject Introspection (Recommended)

- Download: <https://download.gnome.org/sources/gobject-introspection/1.80/gobject-introspection-1.80.1.tar.xz>
- Download MD5 sum: ef1496a7a7abfc31d25b3459ea86ebb
- Download size: 1.0 MB

Patch for Log Level Selection (Optional)

- Optional patch: https://www.linuxfromscratch.org/patches/blfs/12.2/glib-skip_warnings-1.patch

GLib Dependencies

Required

[packaging-24.1](#)

Recommended

[docutils-0.21.2](#), [libxslt-1.1.42](#), and [pcre2-10.44](#)

Optional

[Cairo-1.18.0](#) (for some GObject Introspection tests), [dbus-1.14.10](#) (for some tests), [Fuse-3.16.2](#) and [bindfs](#) (both needed for one test), [GDB-15.1](#) (for bindings), [Gjs-1.80.2](#) (for some GObject Introspection tests), [GTK-Doc-1.34.0](#) (for GObject Introspection documentation), [docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [Gi-DocGen-2024.1](#) (to build API documentation), [glib-networking-2.80.0](#) (for some tests, but this is a circular dependency), [Mako-1.3.5](#) and [Markdown-3.6](#) (both for `g-ir-doc-tool`), and [sysprof](#)

Additional Runtime Dependencies

Quoted directly from the `INSTALL` file: "Some of the mimetype-related functionality in GIO requires the `update-mime-database` and `update-desktop-database` Utilities," which are part of [shared-mime-info-2.4](#) and [desktop-file-utils-0.27](#), respectively. These two utilities are also needed for some tests.

Installation of GLib

If desired, apply the optional patch. In many cases, applications that use this library, either directly or indirectly via other libraries such as [GTK+-3.24.43](#), output numerous warnings when run from the command line. This patch enables the use of an environment variable, `GLIB_LOG_LEVEL`, that suppresses unwanted messages. The value of the variable is a digit that corresponds to:

- 1 Alert
- 2 Critical
- 3 Error
- 4 Warning
- 5 Notice

For instance `export GLIB_LOG_LEVEL=4` will skip output of Warning and Notice messages (and Info/Debug messages if they are turned on). If `GLIB_LOG_LEVEL` is not defined, normal message output will not be affected.

```
patch -Np1 -i ../glib-skip_warnings-1.patch
```

Warning

If a previous version of glib is installed, move the headers out of the way so that later packages do not encounter conflicts:

```
if [ -e /usr/include/glib-2.0 ]; then
    rm -rf /usr/include/glib-2.0.old &&
    mv -vf /usr/include/glib-2.0{,.old}
fi
```

Install GLib by running the following commands:

```
mkdir build &&
cd build &&

meson setup ..           \
    --prefix=/usr          \
    --buildtype=release    \
    -D introspection=disabled \
    -D man-pages=enabled   &&
ninja
```

The GLib test suite requires desktop-file-utils for some tests. However, desktop-file-utils requires GLib in order to compile; therefore, you must first install GLib and then run the test suite.

As the `root` user, install this package for the first time to allow building GObject Introspection:

```
ninja install
```

Build GObject Introspection:

```
tar xf ../../gobject-introspection-1.80.1.tar.xz &&

meson setup gobject-introspection-1.80.1 gi-build \
    --prefix=/usr --buildtype=release &&
ninja -C gi-build
```

To test the results of GObject Introspection, issue: `ninja -C gi-build test`.

As the `root` user, install GObject Introspection for generating the introspection data of GLib libraries (required by various packages using Glib, especially some GNOME packages):

```
ninja -C gi-build install
```

Now generate the introspection data:

```
meson configure -D introspection=enabled &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed 's/glib-2.0/glib-2.80.4/'      \
    -i ../../docs/reference/meson.build &&
meson configure -D documentation=true &&
ninja
```

As the `root` user, install this package again for the introspection data (and optionally, the documentation):

```
ninja install
```

You should now install [desktop-file-utils-0.27](#) and [shared-mime-info-2.4](#) and proceed to run the test suite.

Warning

Do not run the test suite as `root` or some tests will fail unexpectedly and leave some non-FHS-compliant directories in the `/usr` hierarchy.

Note

If you have installed the glib-skip_warnings-1.patch and have the environment variable GLIB_LOG_LEVEL set, unset it before running the tests. It will cause several tests to fail.

To test the results, after having installed the packages, issue: `LC_ALL=C ninja test` as a non-`root` user.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D man-pages=enabled`: This switch causes the build to create and install the package man pages.

Contents

GLib Contents

Installed Programs: `gapplication`, `gdbus`, `gdbus-codegen`, `gi-compile-repository`, `gi-decompile-typelib`, `gi-inspect-typelib` `gio`, `gio-querymodules`, `glib-compile-resources`, `glib-compile-schemas`, `glib-genmarshal`, `glib-gettextize`, `glib-mkenums`, `gobject-query`, `gresource`, `gsettings`, `gtester`, and `gtester-report`

Installed Libraries: `libgio-2.0.so`, `libgirepository-2.0.so`, `libglib-2.0.so`, `libgmodule-2.0.so`, `libgobject-2.0.so`, and `libgthread-2.0.so`

Installed Directories: `/usr/include/gio-unix-2.0`, `/usr/include/glib-2.0`, `/usr/lib/gio`, `/usr/lib/glib-2.0`, `/usr/share/glib-2.0`, and `/usr/share/doc/glib-2.80.4` (optional)

GObject Introspection Contents

Installed Program: `g-ir-annotation-tool`, `g-ir-compiler`, `g-ir-doc-tool` (optional), `g-ir-generate`, `g-ir-inspect`, and `g-ir-scanner`

Installed Libraries: `libgirepository-1.0.so` and `_giscanner.cpython-312-<arch>-linux-gnu.so`

Installed Directories: `/usr/include/gobject-introspection-1.0`, `/usr/lib/girepository-1.0`, `/usr/lib/gobject-introspection`, `/usr/share/gir-1.0`, and `/usr/share/gobject-introspection-1.0`

Short Descriptions

<code>gapplication</code>	can be used to start applications and to send messages to already-running instances of other applications
<code>gdbus</code>	is a simple tool used for working with D-Bus objects
<code>gdbus-codegen</code>	is used to generate code and/or documentation for one or more D-Bus interfaces
<code>gi-compile-repository</code>	converts one or more GIR files into one or more typelib files
<code>gi-decompile-typelib</code>	is a GIR decompiler that uses the repository API
<code>gi-inspect-typelib</code>	is a utility that gives information about a GI typelib
<code>gio</code>	is a utility that makes many GIO features available from the command line
<code>gio-querymodules</code>	is used to create a <code>.giomodule.cache</code> file in the listed directories. This file lists the implemented extension points for each module that has been found
<code>glib-compile-resources</code>	is used to read the resource description from a file and the files that it references to create a binary resource bundle that is suitable for use with the GResource API
<code>glib-compile-schemas</code>	is used to compile all the GSettings XML schema files in a directory into a binary file with the name <code>gschemas.compiled</code> that can be used by GSettings
<code>glib-genmarshal</code>	is a C code marshaller generation utility for GLib closures
<code>glib-gettextize</code>	is a variant of the <code>gettext</code> internationalization utility
<code>glib-mkenums</code>	is a C language enum description generation utility
<code>gobject-query</code>	is a small utility that draws a tree of types
<code>gresource</code>	offers a simple command line interface to GResource
<code>gsettings</code>	offers a simple command line interface to GSettings
<code>gtester</code>	is a test running utility

<code>gtester-report</code>	is a test report formatting utility
<code>libgio-2.0.so</code>	is a library providing useful classes for general purpose I/O, networking, IPC, settings, and other high level application functionality
<code>libgirepository-2.0.so</code>	is a library providing access to typelibs and introspection data which describes C APIs
<code>libglib-2.0.so</code>	is a general-purpose, portable utility library, which provides many useful data types, macros, type conversions, string utilities, file utilities, a mainloop abstraction, and so on
<code>libgmodule-2.0.so</code>	provides portable API for dynamically loading modules
<code>libgobject-2.0.so</code>	provides the GLib base type system and object class
<code>libgthread-2.0.so</code>	is a skeleton library for backwards compatibility; it used to be the GLib thread library but the functionalities has been merged info <code>libglib-2.0</code>
<code>g-ir-annotation-tool</code>	creates or extracts annotation data from GI typelibs
<code>g-ir-compiler</code>	is a counterpart of <code>gi-compile-repository</code> for the old <code>libgirepository-1.0</code> API
<code>g-ir-doc-tool</code>	generates Mallard files that can be viewed with <code>yelp</code> or rendered to HTML with <code>yelp-build</code> from yelp-tools
<code>g-ir-inspect</code>	is a counterpart of <code>gi-inspect-typelib</code> for the old <code>libgirepository-1.0</code> API
<code>g-ir-generate</code>	is a counterpart of <code>gi-decompile-typelib</code> for the old <code>libgirepository-1.0</code> API
<code>g-ir-scanner</code>	is a tool which generates GIR XML files by parsing headers and introspecting GObject based libraries
<code>libgirepository-1.0.so</code>	is a counterpart of <code>libgirepository-2.0</code> with the old 1.0 API

GLibmm-2.66.7

Introduction to GLibmm

The GLibmm package is a set of C++ bindings for GLib.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/glibmm/2.66/glibmm-2.66.7.tar.xz>
- Download MD5 sum: c6edf4cc986adec2a6d21e7423bad7d1
- Download size: 8.4 MB
- Estimated disk space required: 83 MB (with tests)
- Estimated build time: 0.5 SBU (Using parallelism=4; with tests)

GLibmm Dependencies

Required

[GLib-2.80.4](#) and [libsigc++-2.12.1](#)

Optional

[Doxygen-1.12.0](#), [glib-networking-2.80.0](#) (for tests), [GnuTLS-3.8.7.1](#) (for tests), [libxslt-1.1.42](#), and [mm-common](#)

Installation of GLibmm

Install GLibmm by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: libgiomm-2.4.so, libglibmm-2.4.so and libglibmm_generate_extra_defs-2.4.so

Installed Directories: /usr/lib/g{io,lib}mm-2.4 and /usr/include/g{io,lib}mm-2.4

Short Descriptions

libgiomm-2.4.so	contains the GIO API classes
libglibmm-2.4.so	contains the GLib API classes

GLibmm-2.80.0

Introduction to GLibmm

The GLibmm package is a set of C++ bindings for GLib. This version is part of a new API for supporting gtkmm-4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/glibmm/2.80/glibmm-2.80.0.tar.xz>
- Download MD5 sum: 7e58344303cec7cd5f1e812f4c70cb43
- Download size: 9.1 MB
- Estimated disk space required: 97 MB (with tests)
- Estimated build time: 0.6 SBU (Using parallelism=4; with tests)

GLibmm Dependencies

Required

[GLib-2.80.4](#) and [libsigc++-3.6.0](#)

Optional

[Doxygen-1.12.0](#), [glib-networking-2.80.0](#) (for tests), [GnuTLS-3.8.7.1](#) (for tests), [libxslt-1.1.42](#), and [mm-common](#)

Installation of GLibmm

Install GLibmm by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: libgiomm-2.68.so, libglibmm-2.68.so, and libglibmm_generate_extra_defs-2.68.so

Installed Directories: /usr/lib/g{io,lib}mm-2.68 and /usr/include/g{io,lib}mm-2.68

Short Descriptions

libgiomm-2.68.so	contains the GIO API classes
libglibmm-2.68.so	contains the GLib API classes

GMime-3.2.7

Introduction to GMime

The GMime package contains a set of utilities for parsing and creating messages using the Multipurpose Internet Mail Extension (MIME) as defined by the applicable RFCs. See the [GMime web site](#) for the RFCs resourced. This is useful as it provides an API which adheres to the MIME specification as closely as possible while also providing programmers with an extremely easy to use interface to the API functions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gmime/3.2/gmime-3.2.7.tar.xz>
- Download MD5 sum: 7ecd9aa75e0cd2e8668206b1d53df874
- Download size: 2.1 MB
- Estimated disk space required: 25 MB (with tests)
- Estimated build time: 0.3 SBU (with tests)

GMime Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended) and [libgpg-error-1.50](#)

Optional

[DocBook-utils-0.6.14](#), [GPGME-1.23.2](#), [GTK-Doc-1.34.0](#), [libns1-2.0.1](#), [Vala-0.56.17](#), and [Gtk#](#) (requires [Mono](#))

Installation of GMime

Install GMime by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Library: libgmime-3.0.so

Installed Directories: /usr/include/gmime-3.0 and /usr/share/gtk-doc/html/gmime-3.0

Short Descriptions

libgmime-3.0.so	contains API functions used by programs that need to comply to the MIME standards
-----------------	---

Introduction to Gsl

The GNU Scientific Library (GSL) is a numerical library for C and C++ programmers. It provides a wide range of mathematical routines such as random number generators, special functions and least-squares fitting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/gsl/gsl-2.8.tar.gz>
- Download MD5 sum: 182ec03204f164e67238c9116591a37d
- Download size: 8.6 MB
- Estimated disk space required: 223 MB (with tests, without docs)
- Estimated build time: 1.0 SBU (Using parallelism=4; with tests, without docs)

Gsl Dependencies

Optional

[sphinx_rtd_theme-2.0.0](#)

Installation of Gsl

Install Gsl by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

If you have [sphinx_rtd_theme-2.0.0](#) installed, build the documentation with:

```
make html
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you built the documentation, install it (as `root`) with:

```
mkdir /usr/share/doc/gsl-2.8 &&
cp -R doc/_build/html/* /usr/share/doc/gsl-2.8
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: gsl-config, gsl-histogram, and gsl-randist

Installed Libraries: libgslcblas.so and libgsl.so

Installed Directory: /usr/include/gsl and /usr/share/doc/gsl-2.8

Short Descriptions

<code>gsl-config</code>	is a shell script to get the version number and compiler flags of the installed Gsl library
<code>gsl-histogram</code>	is a demonstration program for the GNU Scientific Library that computes a histogram from data taken from stdin
<code>gsl-randist</code>	is a demonstration program for the GNU Scientific Library that generates random samples from various distributions
<code>libgslcblas.so</code>	contains functions that implement a C interface to Basic Linear Algebra Subprograms
<code>libgsl.so</code>	contains functions that provide a collection of numerical routines for scientific computing

gspell-1.12.2

Introduction to gspell

The gspell package provides a flexible API to add spell checking to a GTK+ application.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gspell/1.12/gspell-1.12.2.tar.xz>
- Download MD5 sum: f1e5f02695aee20ba543352889c28ff5
- Download size: 436 KB
- Estimated disk space required: 11 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

gspell Dependencies

Required

[enchant-2.8.2](#), [ICU-75.1](#), and [GTK+-3.24.43](#)

Optional

[GLib-2.80.4](#) (with GObject Introspection), [GTK-Doc-1.34.0](#), [Vala-0.56.17](#), [Valgrind-3.23.0](#), and [Hunspell](#) (for tests)

Installation of gspell

Install gspell by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`. The tests must be run in an X session. One test, `test-checker`, is known to fail if the external package [Hunspell](#) is not installed.

Now, as the `root` user:

```
make install
```

The libtool archive file from this package is referring to the libraries from [ICU-75.1](#). Such a reference may cause packages depending on this package to be unnecessarily linked against the ICU libraries and increase the burden to rebuild packages if ICU is updated to a new major version. As we've explained in [Libtool archive \(.la\) files](#), these libtool archive files are unneeded anyway. So as the `root` user, remove the libtool archive file now:

```
rm -v /usr/lib/libgspell-1.la
```

Contents

Installed Programs: gspell-app1

Installed Libraries: libgspell-1.so

Installed Directories: /usr/include/gspell-1 and /usr/share/gtk-doc/html/gspell-1.0

Short Descriptions

`gspell-app1` checks the spelling of a text entered in a window
`libgspell-1.so` is the gspell API library

highway-1.2.0

Introduction to highway

The highway package contains a C++ library that provides portable SIMD/vector intrinsics.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/google/highway/archive/1.2.0/highway-1.2.0.tar.gz>
- Download MD5 sum: 8b3d090a2d081730b40bca5ae0d65f11
- Download size: 2.1 MB
- Estimated disk space required: 21 MB
- Estimated build time: 0.8 SBU (with parallelism=4)

highway Dependencies

Required

[CMake-3.30.2](#)

Installation of highway

Install highway by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-D BUILD_SHARED_LIBS=ON \
-G Ninja .. &&
ninja
```

This package does come with a test suite, but it requires [gtest](#), which is not in BLFS.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D BUILD_TESTING=OFF`: This parameter disables the test suite from being built because [gtest](#) is not part of BLFS. Without this parameter, CMake will download this package during the configuration process. If you wish to run the tests, install [gtest](#) and then remove this parameter.

`-D BUILD_SHARED_LIBS=ON`: This parameter enables building shared versions of the libraries instead of static versions.

Contents

Installed Programs: None

Installed Libraries: libhwy.so, libhwy_contrib.so, and libhwy_test.so

Installed Directories: /usr/include/hwy and /usr/lib/cmake/hwy

Short Descriptions

libhwy.so	contains functions that provide portable SIMD/vector intrinsics
libhwy_contrib.so	contains several additions to Highway, including a series of dot product, image, math, and sort routines
libhwy_test.so	contains test helpers for Highway

icu-75.1

Introduction to ICU

The International Components for Unicode (ICU) package is a mature, widely used set of C/C++ libraries providing Unicode and Globalization support for software applications. ICU is widely portable and gives applications the same results on all platforms.

Warning

Upgrading this package to a new major version (for example, from 72.1 to 75.1) will require rebuilding of many other packages. If some packages that use the libraries built by icu4c-75 are rebuilt, they will use the new libraries while current packages will use the previous libraries. If the Linux application loader (`/usr/lib/ld-linux-x86-64.so.2`) determines that both the old and new libraries are needed, and a symbol (name of data or function) exists in both versions of the library, all references to the symbol will be resolved to the version appearing earlier in the breadth-first sequence of the dependency graph. This may result in the application failing if the definition of the data or the behavior of the function referred by the symbol differs in two versions. To avoid the issue, users will need to rebuild every package linked to an ICU library as soon as possible once ICU is updated to a new major version.

To determine what external libraries are needed (directly or indirectly) by an application or a library, run:

```
ldd <application or library>
```

or to see only the directly needed libraries:

```
readelf -d <application or library> | grep NEEDED
```

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/unicode-org/icu/releases/download/release-75-1/icu4c-75_1-src.tgz
- Download MD5 sum: a83c1499e508f73ddbc60002f84ea42a
- Download size: 25 MB
- Estimated disk space required: 341 MB (add 45 MB for tests)
- Estimated build time: 0.7 SBU (Using parallelism=4; add 2.0 SBU for tests)

ICU Dependencies

Optional

[LLVM-18.1.7](#) (with Clang), and [Doxygen-1.12.0](#) (for documentation)

Installation of ICU

Note

This package expands to the directory `icu`.

Note

If `clang++` is available, it will be used in the mistaken belief that `g++` might not support C++11, even though `configure` has tested for that. If using `g++` there will be an unnecessary warning at the end of `configure`. Building with `g++` also takes longer than the estimated SBU shown.

Install ICU by running the following commands:

```
cd source
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: derb, escapesrc, genbrk, gencode, gencfu, gencmn, gencnval, gendict, gennorm2, genrb, gensprep, icu-config, icuexportdata, icuinfo, icupkg, makeconv, pkgdata, and uconv

Installed Libraries: libicudata.so, libicui18n.so, libicuio.so, libicutest.so, libicutu.so, and libicuuc.so

Installed Directories: /usr/include/unicode, /usr/lib/icu, and /usr/share/icu

Short Descriptions

derb	disassembles a resource bundle
escapesrc	converts "\u" escaped characters into unicode characters
genbrk	compiles ICU break iteration rules source files into binary data files
gencode	generates C or platform specific assembly code from an ICU data file
gencfu	reads in Unicode confusable character definitions and writes out the binary data
gencmn	generates an ICU memory-mappable data file
gencnval	compiles the converter's aliases file
gendict	compiles word lists into ICU string trie dictionaries
gennorm2	builds binary data files with Unicode normalization data
genrb	compiles a resource bundle
gensprep	compiles StringPrep data from filtered RFC 3454 files
icu-config	outputs ICU build options
icuinfo	outputs configuration information about the current ICU
icupkg	extracts or modifies an ICU .dat archive
makeconv	compiles a converter table
pkgdata	packages data for use by ICU
uconv	converts data from one encoding to another
libicudata.so	is the data library
libicui18n.so	is the internationalization (i18n) library
libicuio.so	is the ICU I/O (unicode stdio) library
libicutest.so	is the test library
libicutu.so	is the tool utility library
libicuuc.so	is the common library

inih-58

Introduction to inih

This package is a simple .INI file parser written in C.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/benhoyt/inih/archive/r58/inih-r58.tar.gz>
- Download MD5 sum: 5c9725320ad2c79e0b1f76568bd0ff24
- Download size: 20 KB
- Estimated disk space required: 724 KB
- Estimated build time: less than 0.1 SBU

Installation of inih

Install inih by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Program: None

Installed Libraries: libinih.so and libINIReader.so

Installed Directories: None

Short Descriptions

`libinih.so` is the inih main library
`libINIReader.so` is the inih parser library

Intel-gmmlib-22.4.1

Introduction to Intel-gmmlib

The Intel-gmmlib package contains the Intel Graphics Memory Management Library, which provides device specific memory and buffer management functions for the Intel Media Driver for VAAPI and the Intel Graphics Computer Runtime for OpenCL (TM).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/intel/gmmlib/archive/intel-gmmlib-22.4.1.tar.gz>
- Download MD5 sum: 37684beaa140a0926967ebd56193cc13
- Download size: 848 KB
- Estimated disk space required: 54 MB
- Estimated build time: 0.5 SBU

Note

The tarball `intel-gmmlib-22.4.1.tar.gz` will extract to the directory `gmmlib-intel-gmmlib-22.4.1`.

Intel-gmmlib Dependencies

Required

[CMake-3.30.2](#)

Installation of Intel-gmmlib

Install Intel-gmmlib by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D BUILD_TYPE=Release \
-G Ninja \
-W no-dev .. &&
ninja
```

The test suite is normally run by `ninja` unless `-D RUN_TEST_SUITE=NO` is passed to `cmake`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: libigdgmm.so

Installed Directories: /usr/include/igdgmm

Short Descriptions

libigdgmm.so contains functions that provide Memory Management functions for Intel Graphics Drivers

Jansson-2.14

Introduction to Jansson

The Jansson package contains a library used to encode, decode, and manipulate JSON data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/akheron/jansson/releases/download/v2.14/jansson-2.14.tar.bz2>
- Download MD5 sum: 3f90473d7d54ebd1cb6a2757396641df
- Download size: 424 KB
- Estimated disk space required: 5.6 MB (add 1.9 MB for tests)
- Estimated build time: 0.1 SBU (with tests)

Installation of Jansson

First fix one of the tests:

```
sed -e "/DT/s;| sort;| sed 's/@@libjansson.*//' &;" \
-i test/suites/api/check-exports
```

Install jansson by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libjansson.so

Installed Directories: None

Short Descriptions

libjansson.so contains an API for encoding, decoding, and manipulating JSON data

JSON-C-0.17

Introduction to JSON-C

The JSON-C implements a reference counting object model that allows you to easily construct JSON objects in C, output them as JSON formatted strings and parse JSON formatted strings back into the C representation of JSON objects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://s3.amazonaws.com/json-c_releases/releases/json-c-0.17.tar.gz
- Download MD5 sum: bad8f5e91b7b2563ee2d507054c70eb2
- Download size: 384 KB
- Estimated disk space required: 7.9 MB
- Estimated build time: 0.4 SBU (with tests)

JSON-C Dependencies

Required

[CMake-3.30.2](#)

Optional (for documentation)

[Doxygen-1.12.0](#) and [Graphviz-12.1.0](#) (for dot tool)

Installation of JSON-C

Install JSON-C by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_STATIC_LIBS=OFF \
      ... &&
make
```

If you have installed [Doxygen-1.12.0](#) and [Graphviz-12.1.0](#), you can build the documentation by running the following command:

```
doxygen doc/Doxyfile
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

If you built the documentation, install it by running the following commands as the `root` user:

```
install -d -vm755 /usr/share/doc/json-c-0.17 &&
install -v -m644 doc/html/* /usr/share/doc/json-c-0.17
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply a higher level of compiler optimizations.

Contents

Installed Programs: None

Installed Libraries: libjson-c.so

Installed Directories: /usr/include/json-c

Short Descriptions

libjson-c.so contains the JSON-C API functions

JSON-GLib-1.8.0

Introduction to JSON GLib

The JSON GLib package is a library providing serialization and deserialization support for the JavaScript Object Notation (JSON) format described by RFC 4627.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/json-glib/1.8/json-glib-1.8.0.tar.xz>
- Download MD5 sum: f1aac2b8a17fd68646653cc4d8426486
- Download size: 156 KB
- Estimated disk space required: 7.8 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

JSON-GLib Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection required if building GNOME)

Optional

[GTK-Doc-1.34.0](#)

Installation of JSON GLib

Install JSON GLib by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=disabled`: Add this option if you have [GTK-Doc-1.34.0](#) installed and do not wish to generate the API documentation.

Contents

Installed Programs: json-glib-format and json-glib-validate

Installed Library: libjson-glib-1.0.so

Installed Directories: /usr/{include,libexec,share{./installed-tests}}/json-glib-1.0} and /usr/share/gtk-doc/html/json-glib

Short Descriptions

<code>json-glib-format</code>	is a simple command line interface to format JSON data
<code>json-glib-validate</code>	is a simple command line interface to validate JSON data
<code>libjson-glib-1.0.so</code>	contains the JSON GLib API functions

keyutils-1.6.3

Introduction to keyutils

Keyutils is a set of utilities for managing the key retention facility in the kernel, which can be used by filesystems, block devices and more to gain and retain the authorization and encryption keys required to perform secure operations.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://git.kernel.org/pub/scm/linux/kernel/git/dhowells/keyutils.git/snapshot/keyutils-1.6.3.tar.gz>
- Download MD5 sum: 6b70b2b381c1b6d9adfaff6d5d3e7c00
- Download size: 136 KB
- Estimated disk space required: 2.6 MB (with tests)
- Estimated build time: less than 0.1 SBU (add 0.4 SBU for tests)

Keyutils Dependencies

Optional

[lsb-tools-0.12](#) (referred by the test suite)

Kernel Configuration

If running the test suite, some tests needs the following kernel features enabled:

```
Security options --->
[*] Enable access key retention support [KEYS]
[*] Large payload keys [BIG_KEYS]
[*] Diffie-Hellman operations on retained keys [KEY_DH_OPERATIONS]

--> Cryptographic API ---> [CRYPTO]
Public-key cryptography --->
<*/M> RSA (Rivest-Shamir-Adleman) [CRYPTO_RSA]
Hashes, digests, and MACs --->
<*/M> SHA-1 [CRYPTO_SHA1]
[*] Asymmetric (public-key cryptographic) key type ---> [ASYMMETRIC_KEY_TYPE]
<> Asymmetric public-key crypto algorithm subtype ...
    ... [ASYMMETRIC_PUBLIC_KEY_SUBTYPE]
# If not built into the kernel, [SYSTEM_TRUSTED_KEYRING] won't show up;
# building as a module won't work:
<> X.509 certificate parser [X509_CERTIFICATE_PARSER]
Certificates for signature checking --->
[*] Provide system-wide ring of trusted keys [SYSTEM_TRUSTED_KEYRING]
[*] Provide a keyring to which extra trustable keys may be added ...
    ... [SECONDARY_TRUSTED_KEYRING]
[*] Provide system-wide ring of blacklisted keys [SYSTEM_BLACKLIST_KEYRING]

Library routines --->
Crypto library routines --->
# If not built into the kernel, [BIG_KEYS] won't show up;
# building as a module won't work:
<> ChaCha20-Poly1305 AEAD support (8-byte nonce library version)
    ... [CRYPTO_LIB_CHACHA20POLY1305]
```

Installation of keyutils

Install keyutils by running the following commands:

```
make
```

Now, as the `root` user:

```
make NO_ARLIB=1 LIBDIR=/usr/lib BINDIR=/usr/bin SBINDIR=/usr/sbin install
```

The test suite can only run after installing this package. To test the results, issue, as the `root` user:

```
make -k test
```

If [lsb-tools-0.12](#) is not installed, the test suite will output some lines complaining the `lsb_release` command not available but it won't affect the test result.

Command Explanations

`NO_ARLIB=1`: This make flag disables installing the static library.

Configuring keyutils

Config Files

`/etc/request-key.conf` and `/etc/request-key.d/*`

Contents

Installed Programs: `keyctl`, `key.dns_resolver`, and `request-key`

Installed Library: `libkeyutils.so`

Installed Directory: `/etc/keyutils`, `/etc/request-key.d`, and `/usr/share/keyutils`

Short Descriptions

<code>keyctl</code>	controls the key management facility with a variety of subcommands
<code>key.dns_resolver</code>	is invoked by <code>request-key</code> on behalf of the kernel when kernel services (such as NFS, CIFS and AFS) need to perform a hostname lookup and the kernel does not have the key cached. It is not ordinarily intended to be called directly
<code>request-key</code>	is invoked by the kernel when the kernel is asked for a key that it doesn't have immediately available. The kernel creates a temporary key and then calls out to this program to instantiate it. It is not intended to be called directly
<code>libkeyutils.so</code>	contains the keyutils library API instantiation

libaio-0.3.113

Introduction to libaio

The libaio package is an asynchronous I/O facility ("async I/O", or "aio") that has a richer API and capability set than the simple POSIX async I/O facility. This library, libaio, provides the Linux-native API for async I/O. The POSIX async I/O facility requires this library in order to provide kernel-accelerated async I/O capabilities, as do applications which require the Linux-native async I/O API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://pagure.io/libaio/archive/libaio-0.3.113/libaio-0.3.113.tar.gz>
- Download MD5 sum: 605237f35de238dfacc83bcae406d95d
- Download size: 48 KB
- Estimated disk space required: 1.0 MB
- Estimated build time: less than 0.1 SBU

Installation of libaio

First, disable the installation of the static library:

```
sed -i '/install.*libaio.a/s/^/#/' src/Makefile
```

Build libaio by running the following command:

```
make
```

To test the results, issue: `make partcheck`.

Now, install the package as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libaio.so

Installed Directories: None

Short Descriptions

libaio.so is the libaio library

libarchive-3.7.4

Introduction to libarchive

The libarchive library provides a single interface for reading/writing various compression formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libarchive/libarchive/releases/download/v3.7.4/libarchive-3.7.4.tar.xz>
- Download MD5 sum: 1bab4c1b443ecf4f23ff9881665e680a
- Download size: 5.2 MB
- Estimated disk space required: 42 MB (add 32 MB for tests)
- Estimated build time: 0.4 SBU (add 0.6 SBU for tests)

libarchive Dependencies

Recommended

[libxml2-2.13.3](#)

Optional

[lzo-2.10](#), [Nettle-3.10](#), and [pcre2-10.44](#)

Installation of libarchive

Install libarchive by running the following commands:

```
./configure --prefix=/usr --disable-static --without-expat &&
make
```

To test the results, issue: `LC_ALL=C.UTF-8 make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--without-expat`: This switch disables using expat for xar format support. Due to a bug the xar reader will loop infinitely if expat is used. The upstream prefers libxml2 for xar support anyway.

`--without-nettle`: This switch sets OpenSSL for crypto support instead of preferred Nettle if both packages are installed.

Contents

Installed Programs: bsdcat, bsdcpio, bsdtar, and bsdunzip

Installed Libraries: libarchive.so

Installed Directories: None

Short Descriptions

<code>bsdcat</code>	expands files to standard output
<code>bsdcpio</code>	is a tool similar to <code>cpio</code>
<code>bsdtar</code>	is a tool similar to GNU <code>tar</code>
<code>bsdunzip</code>	is a tool similar to Info-ZIP <code>unzip</code>
<code>libarchive.so</code>	is a library that can create and read several streaming archive formats

libassuan-3.0.1

Introduction to libassuan

The libassuan package contains an inter process communication library used by some of the other GnuPG related packages. libassuan's primary use is to allow a client to interact with a non-persistent server. libassuan is not, however, limited to use with GnuPG servers and clients. It was designed to be flexible enough to meet the demands of many transaction based environments with non-persistent servers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/libassuan/libassuan-3.0.1.tar.bz2>
- Download MD5 sum: 6f0d239302ae3b8d4aefcb499b137530
- Download size: 580 KB
- Estimated disk space required: 6.5 MB (with tests, add 3.4 MB for pdf documentation)
- Estimated build time: 0.2 SBU (with tests and html documentation)

libassuan Dependencies

Required

[libgpg-error-1.50](#)

Optional

[texlive-20240312](#) (or [install-tl-unx](#))

Installation of libassuan

Install libassuan by running the following commands:

```
./configure --prefix=/usr &&
make &&

make -C doc html &&
makeinfo --html --no-split -o doc/assuan_nochunks.html doc/assuan.texi &&
makeinfo --plaintext -o doc/assuan.txt doc/assuan.texi
```

The above commands build the documentation in html and plaintext formats. If you wish to build alternate formats of the documentation, you must have [texlive-20240312](#) installed and issue the following commands:

```
make -C doc pdf ps
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&

install -v -dm755 /usr/share/doc/libassuan-3.0.1/html &&
install -v -m644 doc/assuan.html/* \
                  /usr/share/doc/libassuan-3.0.1/html &&
```

```
install -v -m644 doc/assuan_nochunks.html \
          /usr/share/doc/libassuan-3.0.1      &&
install -v -m644 doc/assuan.{txt,txxi} \
          /usr/share/doc/libassuan-3.0.1
```

If you built alternate formats of the documentation, install them by running the following commands as the `root` user:

```
install -v -m644 doc/assuan.{pdf,ps,dvi} \
          /usr/share/doc/libassuan-3.0.1
```

Contents

Installed Program: None

Installed Library: libassuan.so

Installed Directory: /usr/share/doc/libassuan-3.0.1

Short Descriptions

`libassuan.so` is an inter process communication library which implements the Assuan protocol

libatasmart-0.19

Introduction to libatasmart

The libatasmart package is a disk reporting library. It only supports a subset of the ATA S.M.A.R.T. functionality.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://0pointer.de/public/libatasmart-0.19.tar.xz>
- Download MD5 sum: 53afe2b155c36f658e121fe6def33e77
- Download size: 248 KB
- Estimated disk space required: 3 MB
- Estimated build time: less than 0.1 SBU

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/libatasmart>

Installation of libatasmart

Install libatasmart by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make docdir=/usr/share/doc/libatasmart-0.19 install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: skdump and sktest

Installed Library: libatasmart.so

Installed Directory: /usr/share/doc/libatasmart-0.19

Short Descriptions

`skdump` is a utility that reports on the status of the disk

sktest is a utility to issue disk tests
libataSMART.so contains the ATA S.M.A.R.T API functions

libatomic_ops-7.8.2

Introduction to libatomic_ops

`libatomic_ops` provides implementations for atomic memory update operations on a number of architectures. This allows direct use of these in reasonably portable code. Unlike earlier similar packages, this one explicitly considers memory barrier semantics, and allows the construction of code that involves minimum overhead across a variety of architectures.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/ivmai/libatomic_ops/releases/download/v7.8.2/libatomic_ops-7.8.2.tar.gz
 - Download MD5 sum: d07b3d8369d7f9efdca59f7501dd1117
 - Download size: 516 KB
 - Estimated disk space required: 6.8 MB (with tests)
 - Estimated build time: 0.2 SBU (with tests)

Installation of libatomic_ops

Install libatomic_ops by running the following commands:

```
./configure --prefix=/usr      \
            --enable-shared \
            --disable-static \
            --docdir=/usr/share/doc/libatomic_ops-7.8.2 &&
make
```

To check the results, issue `make check`.

Now, as the *root* user:

make install

Command Explanations

--enable-shared: This switch enables building of the libatomic_ops shared libraries.

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libatomic_ops.so and libatomic_ops_gpl.so

Installed Directory: /usr/include/libatomic_ops and /usr/share/doc/libatomic_ops-7.8.2

Short Descriptions

`libatomic_ops.so` contains functions for atomic memory operations

libblockdev-3.1.1

Introduction to libblockdev

`libblockdev` is a C library supporting GObject Introspection for manipulation of block devices. It has a plugin-based architecture where each technology (like LVM, Btrfs, MD RAID, Swap,...) is implemented in a separate plugin, possibly with multiple implementations (e.g. using LVM CLI or the new LVM DBus API).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/storaged-project/libblockdev/releases/download/3.1.1-1/libblockdev-3.1.1.tar.gz>
- Download MD5 sum: 28d43f2d6bff85245502a8c4c478c5a3
- Download size: 1.0 MB
- Estimated disk space required: 13 MB
- Estimated build time: 0.1 SBU

libblockdev Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection required for GNOME)

Recommended

[cryptsetup-2.7.4](#), [keyutils-1.6.3](#), [libbytesize-2.11](#), [libnvme-1.10](#), and [LVM2-2.03.26](#)

Optional

[btrfs-progs-6.10.1](#), [GTK-Doc-1.34.0](#), [mdadm-4.3](#), [parted-3.6](#), [volume_key](#), [ndctl](#), and [targetcli](#) (for tests)

Installation of libblockdev

First, work around a problem when using e2fsprogs-1.47.1 or later:

```
find -name Makefile.in -exec sed -i "s/-Werror//" {} \;
```

Install libblockdev by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --with-python3   \
            --without-escrow \
            --without-gtk-doc \
            --without-lvm    \
            --without-lvm_dbus \
            --without-nvdimm \
            --without-tools   &&
make
```

The test suite requires targetcli, which is not a part of BLFS.

Now, as the `root` user:

```
make install
```

Command Explanations

`--without-escrow`: This switch allows building libblockdev without `volume_key` installed.

`--without-lvm --without-lvm_dbus --without-tools`: The combination of these switches allows building libblockdev without [parted-3.6](#) installed. Remove it if you've installed [parted-3.6](#) and wish to build the `lvm-cache-stats` OR `vfat-resize` utilities, or wish libblockdev to support logical volumes.

`--without-btrfs --without-mdraid --without-tools`: The combination of these switches allows building libblockdev without [libbytesize-2.11](#) installed. The `--without-mdraid` option will break [UDisks-2.10.1](#), so don't use it unless you really know what you are doing.

`--without-crypto`: This switch allows building libblockdev without [cryptsetup-2.7.4](#) or [keyutils-1.6.3](#) installed. This switch will break [UDisks-2.10.1](#), so don't use it unless you really know what you are doing.

`--without-dm --without-lvm --without-lvm_dbus --without-mpath`: The combination of these switches allows building libblockdev without [LVM2-2.03.26](#) installed. The `--without-dm` and `--without-mpath` options are not in the instructions provided by the book because [LVM2-2.03.26](#) is required for [cryptsetup-2.7.4](#) and [cryptsetup-2.7.4](#) is a recommended dependency of libblockdev anyway.

--without-nvme: This switch allows building libblockdev without [libnvme-1.10](#) installed. This switch will break [UDisks-2.10.1](#), so don't use it unless you really know what you are doing.

Contents

Installed Programs: lvm-cache-stats and vfat-resize (both optional)

Installed Libraries: libbd_btrfs.so, libbd_crypto.so, libbd_dm.so, libbd_fs.so, libbd_loop.so, libbd_lvm.so (optional), libbd_lvm-dbus.so (optional), libbd_mdraid.so, libbd_mpath.so, libbd_nvme.so, libbd_part.so, libbd_swap.so, libbd_utils.so, and libblockdev.so

Installed Directories: /etc/libblockdev, /usr/include/blockdev, and /usr/share/gtk-doc/html/libblockdev (optional)

Short Descriptions

<code>lvm-cache-stats</code>	prints statistics on caches on LVM Logical Volumes
<code>vfat-resize</code>	resizes <code>vfat</code> file systems

libbytesize-2.11

Introduction to libbytesize

The libbytesize package is a library which facilitates the common operations with sizes in bytes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/storaged-project/libbytesize/releases/download/2.11/libbytesize-2.11.tar.gz>
- Download MD5 sum: b9a24d3a7b576e67cab729195c26f6e5
- Download size: 452 KB
- Estimated disk space required: 3.5 MB
- Estimated build time: less than 0.1 SBU (including tests)

libbytesize Dependencies

Required

[pcre2-10.44](#) and [Pygments-2.18.0](#)

Optional

[GTK-Doc-1.34.0](#), [pocketlint](#) (python module for one test), and [polib](#) (python module for one test)

Installation of libbytesize

Install libbytesize by running the following commands:

```
./configure --prefix=/usr &&  
make
```

If you have the optional python modules installed, the regression tests can be run with: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: bscalculator

Installed Library: libbytesize.so

Installed Directories: /usr/include/bytesize and /usr/lib/python3.12/site-packages/bytesize

Short Descriptions

<code>bscalc</code>	converts from a larger unit, such as MB or TB, back to a value in bytes
<code>libbytesize.so</code>	contains functions used to handle common read/write operations with sizes in bytes

libclc-18.1.7

Introduction to libclc

The libclc package contains an implementation of the library requirements of the OpenCL C programming language, as specified by the OpenCL 1.1 Specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/llvm/llvm-project/releases/download/llvmorg-18.1.7/libclc-18.1.7.src.tar.xz>
- Download MD5 sum: 0c7627f6c82bb63f712e1cc00d89eed7
- Download size: 124 KB
- Estimated disk space required: 380 MB
- Estimated build time: 1.2 SBU (with parallelism=8)

libclc Dependencies

Required

[SPIRV-LLVM-Translator-18.1.3](#)

Installation of libclc

Install libclc by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -G Ninja .. &&
ninja
```

To test the results, issue: `ninja test`. Two tests, `external-calls-clspv--.bc` and `external-calls-clspv64--.bc`, are known to fail due to invalid syntax.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/include/clc and /usr/share/clc

libcloudproviders-0.3.6

Introduction to libcloudproviders

The libcloudproviders package contains a library which provides a DBus API that allows cloud storage sync clients to expose their services.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libcloudproviders/0.3/libcloudproviders-0.3.6.tar.xz>

- Download MD5 sum: f0f994bdc36fdfe9b31e3655b8071599
- Download size: 24 KB
- Estimated disk space required: 1.4 MB
- Estimated build time: less than 0.1 SBU

libcloudproviders Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of libcloudproviders

Install libcloudproviders by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D enable-gtk-doc`: Use this switch if you have [GTK-Doc-1.34.0](#) installed and wish to generate the API documentation.

Contents

Installed Programs: None

Installed Libraries: libcloudproviders.so

Installed Directories: /usr/include/cloudproviders

Short Descriptions

libcloudproviders.so	contains functions that provide a DBus API to allow cloud storage sync clients to expose their services
----------------------	---

libdaemon-0.14

Introduction to libdaemon

The libdaemon package is a lightweight C library that eases the writing of UNIX daemons.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://0pointer.de/lennart/projects/libdaemon/libdaemon-0.14.tar.gz>
- Download MD5 sum: 509dc27107c21bcd9fbf2f95f5669563
- Download size: 332 KB
- Estimated disk space required: 3 MB

- Estimated build time: 0.1 SBU

libdaemon Dependencies

Optional

[Doxygen-1.12.0](#) and [Lynx-2.9.2](#)

Installation of libdaemon

Install libdaemon by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

If you have Doxygen installed and wish to build the API documentation, issue the following command:

```
make -C doc doxygen
```

This package does not come with a test suite.

Now, as the `root` user:

```
make docdir=/usr/share/doc/libdaemon-0.14 install
```

If you built the API documentation, install it using the following commands as the `root` user:

```
install -v -m755 -d /usr/share/doc/libdaemon-0.14/reference/html &&
install -v -m644 doc/reference/html/* /usr/share/doc/libdaemon-0.14/reference/html &&
install -v -m644 doc/reference/man/man3/* /usr/share/man/man3
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libdaemon.so

Installed Directories: /usr/include/libdaemon and /usr/share/doc/libdaemon-0.14

Short Descriptions

libdaemon.so contains the libdaemon API functions

libdisplay-info-0.2.0

Introduction to libdisplay-info

The libdisplay-info package provides a set of high-level and low-level functions to access detailed Extended Display Identification Data (EDID) information.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/emersion/libdisplay-info/-/releases/0.2.0/downloads/libdisplay-info-0.2.0.tar.xz>
- Download MD5 sum: 160d4159a7805823cf0b3b4f86dfa8d4
- Download size: 96 KB
- Estimated disk space required: 2.8 MB
- Estimated build time: less than 0.1 SBU

libdisplay-info Dependencies

Required

[hwdata-0.385](#)

Installation of libdisplay-info

Install libdisplay-info by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, run `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Library: libdisplay-info.so

Installed Directories: /usr/include/libdisplay-info

libgcrypt-1.11.0

Introduction to libgcrypt

The libgcrypt package contains a general purpose crypto library based on the code used in GnuPG. The library provides a high level interface to cryptographic building blocks using an extendable and flexible API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/libgcrypt/libgcrypt-1.11.0.tar.bz2>
- Download MD5 sum: 321c4975cf6a496f0530b65a673f9a4
- Download size: 4.0 MB
- Estimated disk space required: 146 MB (with tests)
- Estimated build time: 0.6 SBU (with documentation; add 0.4 SBU for tests)

libgcrypt Dependencies

Required

[libgpg-error-1.50](#)

Optional

[texlive-20240312](#) (or [install-tl-unx](#))

Installation of libgcrypt

Install libgcrypt by running the following commands:

```
./configure --prefix=/usr &&
make
```

```
make -C doc html
makeinfo --html --no-split -o doc/gcrypt_nochunks.html doc/gcrypt.texi &&
makeinfo --plaintext -o doc/gcrypt.txt doc/gcrypt.texi
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
install -v -dm755 /usr/share/doc/libgcrypt-1.11.0 &&
install -v -m644 README doc/{README.apichanges,fips*,libgcrypt*} \
/usr/share/doc/libgcrypt-1.11.0 &&

install -v -dm755 /usr/share/doc/libgcrypt-1.11.0/html &&
install -v -m644 doc/gcrypt.html/* \
/usr/share/doc/libgcrypt-1.11.0/html &&
install -v -m644 doc/gcrypt_nochunks.html \
/usr/share/doc/libgcrypt-1.11.0 &&
install -v -m644 doc/gcrypt.{txt,txxi} \
/usr/share/doc/libgcrypt-1.11.0
```

Command Explanations

`--with-capabilities`: This option enables libcap2 support. Note that this breaks [cryptsetup-2.7.4](#)

Contents

Installed Programs: `dumpsexp`, `hmac256`, and `mpicalc`

Installed Library: `libgcrypt.so`

Installed Directory: `/usr/share/doc/libgcrypt-1.11.0`

Short Descriptions

<code>dumpsexp</code>	is a debug tool for S-expressions
<code>hmac256</code>	is a standalone HMAC-SHA-256 implementation used to compute an HMAC-SHA-256 authentication code
<code>mpicalc</code>	is a RPN (Reverse Polish Notation) calculator
<code>libgcrypt.so</code>	contains the cryptographic API functions

libgpg-error-1.50

Introduction to libgpg-error

The libgpg-error package contains a library that defines common error values for all GnuPG components.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/libgpg-error/libgpg-error-1.50.tar.bz2>
- Download MD5 sum: 80e905ce798553c766e3384467dc55ea
- Download size: 1.0 MB
- Estimated disk space required: 11 MB (with tests)
- Estimated build time: 0.3 SBU (with tests)

Installation of libgpg-error

Install libgpg-error by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
install -v -m644 -D README /usr/share/doc/libgpg-error-1.50/README
```

Contents

Installed Programs: gpg-error, gpgt-config, and yat2m

Installed Library: libgpg-error.so

Installed Directories: /usr/share/common-lisp/source/gpg-error, /usr/share/libgpg-error, and /usr/share/doc/libgpg-error-1.50

Short Descriptions

<code>gpg-error</code>	is used to determine libgpg-error error codes
<code>gpgt-config</code>	is a <code>pkg-config</code> style tool for querying the information about installed version of libgpg-error
<code>yat2m</code>	extracts man pages from a Texinfo source
<code>libgpg-error.so</code>	contains the libgpg-error API functions

libgsf-1.14.52

Introduction to libgsf

The libgsf package contains a library used for providing an extensible input/output abstraction layer for structured file formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgsf/1.14/libgsf-1.14.52.tar.xz>
- Download MD5 sum: c64b4a29dd2e67749313f3917a12a709
- Download size: 692 KB
- Estimated disk space required: 14 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/libgsf-1.14.52-upstream_fixes-1.patch

libgsf Dependencies

Required

[GLib-2.80.4](#) and [libxml2-2.13.3](#)

Recommended

[gdk-pixbuf-2.42.12](#) (To build `gsf-office-thumbnailer`)

Optional

[GTK-Doc-1.34.0](#), [p7zip-17.04](#) (for tests), [UnZip-6.0](#) (for tests), and [Valgrind-3.23.0](#) (for tests)

Installation of libgsf

First, fix building libgsf with libxml2-2.13:

```
patch -Np1 -i ../../libgsf-1.14.52-upstream_fixes-1.patch
```

Install libgsf by running the following commands:

```
autoreconf
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`. Two tests related to [Valgrind-3.23.0](#) are known to fail.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: gsf, gsf-office-thumbnailer, and gsf-vba-dump

Installed Library: libgsf-1.so

Installed Directories: /usr/include/libgsf-1, /usr/share/gtk-doc/html/gsf and /usr/share/thumbailers

Short Descriptions

<code>gsf</code>	is a simple archive utility, somewhat similar to tar(1) .
<code>gsf-office-thumbnailer</code>	is used internally by GNOME applications such as Nautilus to generate thumbnails of several types of office application files
<code>gsf-vba-dump</code>	is used to extract Visual Basic for Applications macros from files
<code>libgsf-1.so</code>	contains the libgsf API functions

libgudev-238

Introduction to libgudev

The libgudev package contains GObject bindings for libudev.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgudev/238/libgudev-238.tar.xz>
- Download MD5 sum: 46da30a1c69101c3a13fa660d9ab7b73
- Download size: 32 KB
- Estimated disk space required: 2.0 MB
- Estimated build time: less than 0.1 SBU

Required

[GLib-2.80.4](#) (GObject Introspection required for GNOME)

Optional

[GTK-Doc-1.34.0](#) and [umockdev-0.18.3](#) (for testing)

Installation of libgudev

Install libgudev by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this option if GTK-Doc is installed and you wish to build and install the API documentation.

Contents

Installed Program: None

Installed Libraries: libgudev-1.0.so

Installed Directories: /usr/include/gudev-1.0 and /usr/share/gtk-doc/html/gudev

Short Descriptions

`libgudev-1.0.so` is a GObject-based wrapper library for libudev

libgusb-0.4.9

Introduction to libgusb

The libgusb package contains the GObject wrappers for libusb-1.0 that makes it easy to do asynchronous control, bulk and interrupt transfers with proper cancellation and integration into a mainloop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/hughsie/libgusb/releases/download/0.4.9/libgusb-0.4.9.tar.xz>
- Download MD5 sum: 354a3227334991ea4e924843c144bd82
- Download size: 52 KB
- Estimated disk space required: 5.0 MB (with tests and docs)
- Estimated build time: 0.1 SBU (with tests and docs)

libgusb Dependencies

Required

[JSON-GLib-1.8.0](#) and [libusb-1.0.27](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [hwdata-0.385](#) (for `usb.ids` data file, which is also required for the tests), and [Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#) and [umockdev-0.18.3](#) (for full test coverage)

Installation of libgusb

Install libgusb by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
```

```
-D docs=false      &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed -E "/output|install_dir/s/('libgusb')/\\1-0.4.9/'/" \
-i ../docs/meson.build
meson configure -D docs=true
&&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D docs=false`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

Contents

Installed Programs: gusbcmd

Installed Library: libgusb.so

Installed Directories: /usr/include/gusb-1 and /usr/share/doc/libgusb-0.4.9 (if gi-docgen is used)

Short Descriptions

`gusbcmd` is a debugging tool for the libgusb library
`libgusb.so` contains the libgusb API functions

libical-3.0.18

Introduction to libical

The libical package contains an implementation of the iCalendar protocols and data formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libical/libical/releases/download/v3.0.18/libical-3.0.18.tar.gz>
- Download MD5 sum: cc790ea49701df0335d42b89d57c04bc
- Download size: 888 KB
- Estimated disk space required: 24 MB (with tests)
- Estimated build time: 0.1 SBU (using parallelism=4; with tests)

libical Dependencies

Required

[CMake-3.30.2](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#) (both required for GNOME)

Optional

[Doxygen-1.12.0](#) (for the API documentation), [Graphviz-12.1.0](#) (for the API documentation), [GTK-Doc-1.34.0](#) (for the API documentation), [ICU-75.1](#), [PyGObject-3.48.2](#) (for some tests), and [Berkeley DB](#) (deprecated)

Installation of libical

If [ICU-75.1](#) is installed, fix an incompatibility with ICU 75 or later:

```
sed -i '/getKeywordValuesForLocale/s/NULL//\' src/libical/icalrecur.c
```

Install libical by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D SHARED_ONLY=yes \
      -D ICAL_BUILD_DOCS=false \
      -D ICAL_BUILD_EXAMPLES=false \
      -D GOBJECT_INTROSPECTION=true \
      -D ICAL_GLIB_VAPI=true \
      .. &&
make
```

If you have [Doxygen-1.12.0](#), [Graphviz-12.1.0](#), and [GTK-Doc-1.34.0](#) installed and wish to build the API documentation, you should remove the `-D ICAL_BUILD_DOCS=false` switch and issue:

```
make docs
```

To test the results, issue: `make test`. One test named `icalrecurtest-r` is known to fail.

Now, as the `root` user:

```
make install
```

If you have built the API documentation, install by issuing, as `root` user:

```
install -vdm755 /usr/share/doc/libical-3.0.18/html &&
cp -vr apidocs/html/* /usr/share/doc/libical-3.0.18/html
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply higher level of the compiler optimizations.

`-D SHARED_ONLY=yes`: This switch is used in order to only build the shared libraries.

`-D ICAL_BUILD_DOCS=false`: This switch prevents building the GTK documentation. Remove if you want to build the documentation.

`-D ICAL_BUILD_EXAMPLES=false`: This switch prevents building examples. Remove if you want to build them.

`-D GOBJECT_INTROSPECTION=true`: This switch is used to generate GObject metadata bindings.

`-D ICAL_GLIB_VAPI=true`: This switch is used in order to build bindings for [Vala-0.56.17](#).

`-D USE_BUILTIN_TZDATA=yes`: This switch is used in order to build using your own timezone data.

Contents

Installed Programs: None

Installed Libraries: libical_cxx.so, libical.so, libical-glib.so, libicalss_cxx.so, libicalss.so, and libicalvcal.so

Installed Directory: /usr/include/libical, /usr/include/libical-glib, /usr/lib/cmake/LibIcal, /usr/libexec/libical, /usr/share/gtk-doc/html/libical-glib (optional), and /usr/share/doc/libical-3.0.18/html

Short Descriptions

libical.so	contains the libical API functions
libical_cxx.so	contains the libical C++ bindings

libical-glib.so	contains the libical glib bindings
libicalss.so	is a library that allows you to store iCal component data to disk in a variety of ways
libicalss_cxx.so	contains the libicalss C++ bindings
libicalvcal.so	is a vCard/vCalendar C interface

libidn-1.42

Introduction to libidn

libidn is a package designed for internationalized string handling based on the [Stringprep](#), [Punycode](#) and [IDNA](#) specifications defined by the Internet Engineering Task Force (IETF) Internationalized Domain Names (IDN) working group, used for internationalized domain names. This is useful for converting data from the system's native representation into UTF-8, transforming Unicode strings into ASCII strings, allowing applications to use certain ASCII name labels (beginning with a special prefix) to represent non-ASCII name labels, and converting entire domain names to and from the ASCII Compatible Encoding (ACE) form.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/libidn/libidn-1.42.tar.gz>
- Download MD5 sum: fe061a95ae23979150a692d102dce4ad
- Download size: 2.5 MB
- Estimated disk space required: 19 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libidn Dependencies

Optional

[Emacs-29.4](#), [GTK-Doc-1.34.0](#), [OpenJDK-22.0.2](#), [Valgrind-3.23.0](#), and [Mono](#)

Installation of libidn

Install libidn by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, run:

```
pushd tests &&
make check &&
popd
```

Now, as the `root` user:

```
make install &&

find doc -name "Makefile*" -delete      &&
rm -rf doc/{gdoc,idn.1,stamp-vti,man,txi} &&
mkdir -v      /usr/share/doc/libidn-1.42    &&
cp -r -v doc/* /usr/share/doc/libidn-1.42
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

`--enable-java`: Use this switch to enable building the Java implementation of libidn. Note that [OpenJDK-22.0.2](#) must be installed to use this option.

Contents

Installed Program: idn
Installed Library: libidn.so
Installed Directories: /usr/share/doc/libidn-1.42 and /usr/share/gtk-doc/html/libidn

Short Descriptions

idn is a command line interface to the internationalized domain name library
libidn.so contains a generic Stringprep implementation that does Unicode 3.2 NFKC normalization, mapping and prohibition of characters, and bidirectional character handling. Profiles for Nameprep, iSCSI, SASL and XMPP are included as well as support for Punycode and ASCII Compatible Encoding (ACE) via IDNA. A mechanism to define Top-Level Domain (TLD) specific validation tables, and to compare strings against those tables, as well as default tables for some TLDs are included

libidn2-2.3.7

Introduction to libidn2

libidn2 is a package designed for internationalized string handling based on standards from the Internet Engineering Task Force (IETF)'s IDN working group, designed for internationalized domain names.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/libidn/libidn2-2.3.7.tar.gz>
- Download MD5 sum: de2818c7dea718a4f264f463f595596b
- Download size: 2.1 MB
- Estimated disk space required: 21 MB (add 3 MB for tests)
- Estimated build time: 0.1 SBU (add 0.6 SBU for tests)

libidn2 Dependencies

Recommended

[libunistring-1.2](#)

Optional

[git-2.46.0](#) and [GTK-Doc-1.34.0](#)

Installation of libidn2

Install libidn2 by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Program: idn2
Installed Library: libidn2.so
Installed Directory: /usr/share/gtk-doc/html/libidn2

Short Descriptions

<code>idn2</code>	is a command line interface to the internationalized domain library
<code>libidn2.so</code>	contains a generic Stringprep implementation used for internationalized string handling

libksba-1.6.7

Introduction to Libksba

The Libksba package contains a library used to make X.509 certificates as well as making the CMS (Cryptographic Message Syntax) easily accessible by other applications. Both specifications are building blocks of S/MIME and TLS. The library does not rely on another cryptographic library but provides hooks for easy integration with Libgcrypt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/libksba/libksba-1.6.7.tar.bz2>
- Download MD5 sum: 7e736de467b67c7ea88de746c31ea12f
- Download size: 692 KB
- Estimated disk space required: 9.4 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Libksba Dependencies

Required

[libgpg-error-1.50](#)

Optional

[Valgrind-3.23.0](#)

Installation of Libksba

Install Libksba by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: None

Installed Library: libksba.so

Installed Directory: None

Short Descriptions

<code>libksba.{so,a}</code>	contains the cryptographic API functions
-----------------------------	--

liblinear-247

Introduction to liblinear

This package provides a library for learning linear classifiers for large scale applications. It supports Support Vector Machines (SVM) with L2 and L1 loss, logistic regression, multi class classification and also Linear Programming Machines

(L1-regularized SVMs). Its computational complexity scales linearly with the number of training examples making it one of the fastest SVM solvers around.

This package is known to build and work properly using an LFS 12.2 platform.

Note

After updating this package from liblinear-1.96 or earlier, you need to reinstall [Nmap-7.95](#), in order to link to the new library.

Package Information

- Download (HTTP): <https://github.com/cjlin1/liblinear/archive/v247/liblinear-247.tar.gz>
- Download MD5 sum: 0c81dd2b2cabc4fba85f61b9dc51c4a3
- Download size: 560 KB
- Estimated disk space required: 1.6 MB
- Estimated build time: less than 0.1 SBU

Installation of liblinear

Install liblinear by running the following commands:

```
make lib
```

This package does not come with a test suite.

Now, as the `root` user:

```
install -vm644 linear.h /usr/include &&
install -vm755 liblinear.so.5 /usr/lib &&
ln -sfv liblinear.so.5 /usr/lib/liblinear.so
```

Contents

Installed Programs: None

Installed Library: liblinear.so

Installed Directories: None

Short Descriptions

`liblinear.so` is a large linear classification library

libmbim-1.26.4

Introduction to libmbim

The libmbim package contains a GLib-based library for talking to WWAN modems and devices which speak the Mobile Interface Broadband Model (MBIM) protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/libmbim/libmbim-1.26.4.tar.xz>
- Download MD5 sum: 58dea20cad346f31d2873b68385a9973
- Download size: 553 KB
- Estimated disk space required: 33 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libmbim Dependencies

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#) and [help2man](#)

Installation of libmbbim

Install libmbbim by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: mbimcli and mbim-network

Installed Libraries: libmbbim-glib.so

Installed Directories: /usr/include/libmbbim-glib and /usr/share/gtk-doc/html/libmbbim-glib

Short Descriptions

<code>mbimcli</code>	is an utility used to control MBIM devices
<code>mbim-network</code>	is an utility used for simple network management of MBIM devices
<code>libmbbim-glib.so</code>	contains API functions for talking to WWAN modems and devices which speak the Mobile Interface Broadband Model (MBIM) protocol

libnvme-1.10

Introduction to libnvme

The libnvme package is a library which provides type definitions for NVMe specification structures, enumerations, and bit fields, helper functions to construct, dispatch, and decode commands and payloads, and utilities to connect, scan, and manage NVMe devices on a Linux system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/linux-nvme/libnvme/archive/v1.10/libnvme-1.10.tar.gz>
- Download MD5 sum: 0d43167362a119f8a621607b7ee20c51
- Download size: 704 KB
- Estimated disk space required: 21 MB
- Estimated build time: less than 0.1 SBU (including tests)

libnvme Dependencies

Optional

[JSON-C-0.17](#), [keyutils-1.6.3](#), and [SWIG-4.2.1](#)

Installation of libnvme

Install libnvme by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D libdbus=auto .. &&
ninja
```

To run the tests, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Library: libnvme.so and libnvme-mi.so

Installed Directories: /usr/include/nvme

Short Descriptions

`libnvme.so` contains functions used to handle operations relating to NVMe devices.

libpaper-2.2.5

Introduction to libpaper

This package is intended to provide a simple way for applications to take actions based on a system or user-specified paper size.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/rrthomas/libpaper/releases/download/v2.2.5/libpaper-2.2.5.tar.gz>
- Download MD5 sum: 4b914c3ca6d8277260fce418e12bae6b
- Download size: 1.1 MB
- Estimated disk space required: 11 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Installation of libpaper

Install libpaper by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --disable-static \
            --docdir=/usr/share/doc/libpaper-2.2.5 &&
make
```

The test suite of this package does not work with a configuration using the standard installation location.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Configuring libpaper

Configuration Information

The paper size is automatically determined from the system locale; see `LC_PAPER` in [locale\(7\)](#). If you want to override it, create a `papersize` file in the user configuration directory. For example:

```
mkdir -pv ~/.config &&
echo "a4" > ~/.config/papersize
```

If you want to override the paper size globally (for all users), set up the `PAPERSIZE` environment variable. For example:

```
echo "PAPERSIZE=a4" > /etc/profile.d/libpaper.sh
```

You may use a different paper size, such as `letter`.

Contents

Installed Programs: `paper` and `paperconf`

Installed Library: `libpaper.so`

Installed Directories: `/usr/share/doc/libpaper-2.2.5`

Short Descriptions

<code>paper</code>	prints paper configuration information
<code>paperconf</code>	prints paper configuration information in a compatibility mode
<code>libpaper.so</code>	contains functions for interrogating the paper library

libportal-0.7.1

Introduction to libportal

The libportal package provides a library that contains GIO-style async APIs for most Flatpak portals.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/flatpak/libportal/releases/download/0.7.1/libportal-0.7.1.tar.xz>
- Download MD5 sum: f94be41e8495ae7d6aaed046481daa61
- Download size: 76 KB
- Estimated disk space required: 3.0 MB (with tests)
- Estimated build time: 0.1 SBU (add 0.3 SBU for tests)

Additional Downloads

- Optional patch (required for running the test suite):
https://www.linuxfromscratch.org/patches/blfs/12.2/libportal-0.7.1-testsuite_fix-1.patch

libportal Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection)

Recommended

[GTK+-3.24.43](#) and [GTK-4.14.5](#)

Recommended (Runtime)

To make this package really useful, install [xdg-desktop-portal-1.18.2](#), [xdg-desktop-portal-gtk-1.15.1](#), [xdg-desktop-portal-gnome-46.2](#) (if running a GNOME desktop environment), and [xdg-desktop-portal-lxqt-1.0.2](#) (if running a LXQt desktop environment). They are not needed if only using this package to satisfy a build dependency.

Optional

[Gi-DocGen-2024.1](#) (for documentation), [dbusmock-0.32.1](#) and [pytest-8.3.2](#) (for testing), [qt5-components-5.15.14](#) (for the Qt5 version of libportal), and [Vala-0.56.17](#)

Installation of libportal

Warning

If a previous version of libportal is installed, move the headers out of the way so that later packages do not encounter conflicts (as the `root` user):

```
if [ -e /usr/include/libportal ]; then
    rm -rf /usr/include/libportal.old &&
    mv -vf /usr/include/libportal{,.old}
fi
```

If running the test suite, apply a patch to fix several test failures with python-dbusmock 0.30.0 or later:

```
patch -Npl -i ../libportal-0.7.1-testsuite_fix-1.patch
```

Install libportal by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D docs=false .. &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/output/s/-1/-0.7.1/" -i ../doc/meson.build &&
meson configure -D docs=true &&
ninja
```

To test the results, issue: `ninja test`. Note that additional `ibus-daemon` processes may need to be killed after the tests are run.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D docs=false`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

Contents

Installed Programs: None

Installed Libraries: libportal.so, libportal-gtk3.so, libportal-gtk4.so, and libportal-qt5.so

Installed Directories: /usr/include/libportal and /usr/share/gtk-doc/html/libportal

Short Descriptions

libportal.so	provides GIO-style async APIs for most Flatpak portals
libportal-gtk3.so	provides GTK+-3 specific functions for interacting with Flatpak portals
libportal-gtk4.so	provides GTK-4 specific functions for interacting with Flatpak portals
libportal-qt5.so	provides Qt5 specific functions for interacting with Flatpak portals

Introduction to libptytty

The libptytty package provides a library that allows for OS independent and secure pty/tty and utmp/wtmp/lastlog handling.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://dist.schmorp.de/libptytty/libptytty-2.0.tar.gz>
- Download MD5 sum: 2a7f3f3c0d3ef71902da745dc7959529
- Download size: 48 KB
- Estimated disk space required: 1.4 MB
- Estimated build time: less than 0.1 SBU

libptytty Dependencies

Required

[CMake-3.30.2](#)

Installation of libptytty

Install libptytty by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D PT_UTMP_FILE:STRING=/run/utmp \
      ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libptytty.so

Installed Directories: None

Short Descriptions

libptytty.so provides for OS independent and secure pty/tty and utmp/wtmp/lastlog handling

libqalculate-5.2.0

Introduction to libqalculate

The libqalculate package contains a library that provides functions for a multi-purpose calculator.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Qalculate/libqalculate/releases/download/v5.2.0/libqalculate-5.2.0.tar.gz>
- Download MD5 sum: d378112794a24d6af1092e0c25f0ff97
- Download size: 2.6 MB
- Estimated disk space required: 156 MB (with tests)

- Estimated build time: 0.9 SBU (With tests; both using parallelism=8)

libqalculate Dependencies

Required

[cURL-8.9.1](#), [ICU-75.1](#), and [libxml2-2.13.3](#)

Optional

[Doxygen-1.12.0](#) and [gnuplot](#)

Installation of libqalculate

Install libqalculate by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --docdir=/usr/share/doc/libqalculate-5.2.0 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

The libtool archive file from this package is referring to the libraries from [ICU-75.1](#). Such a reference may cause packages depending on this package to be unnecessarily linked against the ICU libraries and increase the burden to rebuild packages if ICU is updated to a new major version. As we've explained in [Libtool archive \(.la\) files](#), these libtool archive files are unneeded anyway. So as the `root` user, remove the libtool archive file now:

```
rm -v /usr/lib/libqalculate.la
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: qalc

Installed Library: libqalculate.so

Installed Directories: /usr/include/libqalculate, /usr/share/doc/libqalculate-5.2.0, and /usr/share/qalculate

Short Descriptions

<code>qalc</code>	is a powerful and easy to use command line calculator
<code>libqalculate.so</code>	contains the libqalculate API functions

libqmi-1.30.8

Introduction to libqmi

The libqmi package contains a GLib-based library for talking to WWAN modems and devices which speak the Qualcomm MSM Interface (QMI) protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/libqmi/libqmi-1.30.8.tar.xz>
- Download MD5 sum: 5ec8838914f80e1dfa4d2fa8cc2f186d
- Download size: 1.2 MB
- Estimated disk space required: 151 MB (with tests)

- Estimated build time: 0.3 SBU (Using parallelism=4; with tests)

libqmi Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended) and [libgudev-238](#)

Recommended

[libmbim-1.26.4](#)

Optional

[GTK-Doc-1.34.0](#), [help2man](#), and [libqrtr-glib](#)

Installation of libqmi

Install libqmi by running the following commands:

```
PYTHON=python3 ./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

`--disable-mbim-qmux`: This switch disables support for using a MBIM control device for QMI messages. Use this switch if you did not install libmbim.

Contents

Installed Programs: qmicli, qmi-firmware-update, and qmi-network

Installed Libraries: libqmi-glib.so

Installed Directories: /usr/include/libqmi-glib and /usr/share/gtk-doc/html/libqmi-glib

Short Descriptions

<code>qmicli</code>	is an utility used to control QMI devices
<code>qmi-firmware-update</code>	is an utility used to perform firmware updates on QMI devices
<code>qmi-network</code>	is an utility used for simple network management of QMI devices
<code>libqmi-glib.so</code>	contains API functions for talking to WWAN modems and devices which speak the Qualcomm MSM Interface (QMI) protocol

libseccomp-2.5.5

Introduction to libseccomp

The libseccomp package provides an easy to use and platform independent interface to the Linux kernel's syscall filtering mechanism.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/seccomp/libseccomp/releases/download/v2.5.5/libseccomp-2.5.5.tar.gz>
- Download MD5 sum: c27a5e43cae1e89e6ebfedeea734c9b4
- Download size: 628 KB
- Estimated disk space required: 7.1 MB (additional 4.7 MB for tests)
- Estimated build time: less than 0.1 SBU (additional 2.9 SBU for tests)

libseccomp Dependencies

Optional

[Which-2.21](#) (needed for tests), [Valgrind-3.23.0](#), and [LCOV](#)

Installation of libseccomp

Install libseccomp by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: scmp_sys_resolver

Installed Library: libseccomp.so

Installed Directories: None

Short Descriptions

<code>scmp_sys_resolver</code>	is used to resolve system calls for applications
<code>libseccomp.so</code>	contains API functions for translating syscalls

libsigc++-2.12.1

Introduction to libsigc++

The libsigc++ package implements a typesafe callback system for standard C++.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libsigc++/2.12/libsigc++-2.12.1.tar.xz>
- Download MD5 sum: 891f1b2dbaebd8007eda8c639bbe1149
- Download size: 4.8 MB
- Estimated disk space required: 27 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libsigc++ Dependencies

Recommended

[Boost-1.86.0](#) and [libxslt-1.1.42](#)

Optional

[DocBook-utils-0.6.14](#), [docbook-xml-5.0](#), [Doxygen-1.12.0](#), [fop-2.9](#), and [mm-common](#)

Installation of libsigc++

Install libsigc++ by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D build-documentation=true`: Use this switch if [Doxygen-1.12.0](#) is installed and you wish to build and install the API documentation.

Contents

Installed Programs: None

Installed Library: libsigc-2.0.so

Installed Directories: /usr/{include,lib}/sigc++-2.0 and /usr/share/{devhelp/books,doc}/libsigc++-2.0 (if the documentation is enabled)

Short Descriptions

`libsigc-2.0.so` provides a typesafe callback system for standard C++

libsigc++-3.6.0

Introduction to libsigc++

The libsigc++ package implements a typesafe callback system for standard C++. This version is part of a new API for supporting gtkmm-4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libsigc++/3.6/libsigc++-3.6.0.tar.xz>
- Download MD5 sum: b7205d5465ac15fb0c781d39b4011be
- Download size: 972 KB
- Estimated disk space required: 12 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

libsigc++ Dependencies

Recommended

[Boost-1.86.0](#) and [libxslt-1.1.42](#)

Optional

Installation of `libsigc++`

Install `libsigc++` by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D build-documentation=true`: Use this switch if [Doxygen-1.12.0](#) is installed and you wish to build and install the API documentation.

Contents

Installed Programs: None

Installed Library: `libsigc-3.0.so`

Installed Directories: `/usr/{include,lib}/sigc++-3.0` and `/usr/share/{devhelp/books,doc}/libsigc++-3.0` (if the documentation is enabled)

Short Descriptions

`libsigc-3.0.so` provides a typesafe callback system for standard C++

libsigsegv-2.14

Introduction to `libsigsegv`

`libsigsegv` is a library for handling page faults in user mode. A page fault occurs when a program tries to access to a region of memory that is currently not available. Catching and handling a page fault is a useful technique for implementing pageable virtual memory, memory-mapped access to persistent databases, generational garbage collectors, stack overflow handlers, and distributed shared memory.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/libsigsegv/libsigsegv-2.14.tar.gz>
- Download MD5 sum: 63a2b35f11b2fbccc3d82f9e6c6af58
- Download size: 456 KB
- Estimated disk space required: 3.6 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Installation of `libsigsegv`

Install `libsigsegv` by running the following commands:

```
./configure --prefix=/usr \
--enable-shared \
--disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-shared`: This switch ensures that shared libraries are compiled.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: `libsigsegv.so`

Installed Directories: None

Short Descriptions

`libsigsegv.so` is a library for handling page faults in user mode

libssh2-1.11.0

Introduction to libssh2

Libssh2 package is a client-side C library implementing the SSH2 protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.libssh2.org/download/libssh2-1.11.0.tar.gz>
- Download MD5 sum: a01d543fd891ca48fe47726540d50b17
- Download size: 1.1 MB
- Estimated disk space required: 13 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/libssh2-1.11.0-security_fixes-1.patch

libssh2 Dependencies

Optional

[CMake-3.30.2](#) (can be used instead of the `configure` script), [libgcrypt-1.11.0](#) (can be used instead of OpenSSL), [OpenSSH-9.8p1](#) (for some tests), and [Docker](#) (for some tests)

Installation of libssh2

First, fix a critical security vulnerability in libssh2:

```
patch -Np1 -i ../../libssh2-1.11.0-security_fixes-1.patch
```

If you want to test the package, exclude the tests requiring a static library, and remove the reference to an insecure algorithm removed since OpenSSH 9.8:

```
sed -E '/^DOCKER_TEST/,/^SSHD_TEST/s/test_(auth_keyboard_info.* |hostkey |simple)}/${NOTHING}/' \
    -i tests/Makefile.inc &&
autoreconf -fi &&
sed 's/ssh-dss,//' -i tests/openssh_server/sshd_config
```

Install libssh2 by running the following commands:

```
./configure --prefix=/usr      \
            --disable-docker-tests \
            --disable-static      &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-docker-tests`: Disable the tests requiring Docker, which is beyond the scope of BLFS. Remove this switch if you've installed Docker and you wish to run the test suite.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libssh2.so

Installed Directories: None

Short Descriptions

libssh2.so contains functions to use the SSH2 protocol

libstatgrab-0.92.1

Introduction to libstatgrab

This is a library that provides cross platform access to statistics about the system on which it's run. It's written in C and presents a selection of useful interfaces which can be used to access key system statistics. The current list of statistics includes CPU usage, memory utilisation, disk usage, process counts, network traffic, disk I/O, and more.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.mirrorservice.org/sites/ftp.i-scream.org/pub/i-scream/libstatgrab/libstatgrab-0.92.1.tar.gz>
- Download MD5 sum: af685494e985229e0ac46365bc0cd50e
- Download size: 800 KB
- Estimated disk space required: 7.6 MB (with tests)
- Estimated build time: 0.1 SBU (add 1.5 SBU for tests)

libstatgrab Dependencies

Optional

[log4cplus-2.1.1](#)

Installation of libstatgrab

Install libstatgrab by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --docdir=/usr/share/doc/libstatgrab-0.92.1 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: saidar, statgrab, statgrab-make-mrtg-config, and statgrab-make-mrtg-index

Installed Library: libstatgrab.so

Installed Directories: /usr/share/doc/libstatgrab-0.92.1

Short Descriptions

saidar	is a curses-based tool for viewing system statistics
statgrab	is a sysctl-style interface to system statistics
statgrab-make-mrtg-config	generates MRTG configuration
statgrab-make-mrtg-index	generates an XHTML index page from MRTG configuration files or stdin
libstatgrab.so	contains the libstatgrab API functions

libtasn1-4.19.0

Introduction to libtasn1

libtasn1 is a highly portable C library that encodes and decodes DER/BER data following an ASN.1 schema.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/libtasn1/libtasn1-4.19.0.tar.gz>
- Download MD5 sum: f701ab57eb8e7d9c105b2cd5d809b29a
- Download size: 1.7 MB
- Estimated disk space required: 17 MB (with tests)
- Estimated build time: 0.8 SBU (with tests)

libtasn1 Dependencies

Optional

[GTK-Doc-1.34.0](#) and [Valgrind-3.23.0](#)

Installation of libtasn1

Install libtasn1 by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you did not pass the `--enable-gtk-doc` parameter to the `configure` script, you can install the API documentation using the following command as the `root` user:

```
make -C doc/reference install-data-local
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

--enable-gtk-doc: This parameter is normally used if GTK-Doc is installed and you wish to rebuild and install the API documentation. It is broken for this package due to the use of a long deprecated gtk-doc program that is no longer available.

Contents

Installed Programs: asn1Coding, asn1Decoding and asn1Parser

Installed Library: libtasn1.so

Installed Directory: /usr/share/gtk-doc/html/libtasn1

Short Descriptions

asn1Coding	is an ASN.1 DER encoder
asn1Decoding	is an ASN.1 DER decoder
asn1Parser	is an ASN.1 syntax tree generator for libtasn1
libtasn1.so	is a library for Abstract Syntax Notation One (ASN.1) and Distinguish Encoding Rules (DER) manipulation

libunistring-1.2

Introduction to libunistring

libunistring is a library that provides functions for manipulating Unicode strings and for manipulating C strings according to the Unicode standard.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/libunistring/libunistring-1.2.tar.xz>
- Download MD5 sum: 6b4ea63617bf09d76e5234379e75e7f9
- Download size: 2.4 MB
- Estimated disk space required: 54 MB (add 56 MB for tests)
- Estimated build time: 0.5 SBU (add 0.3 SBU for tests; both using parallelism=4)

libunistring Dependencies

Optional

[texlive-20240312](#) (or [install-tl-unx](#)) (to rebuild the documentation)

Installation of libunistring

Install libunistring by running the following commands:

```
./configure --prefix=/usr \
            \
            \
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libunistring.so

Installed Directory: /usr/include/unistring and /usr/share/doc/libunistring-1.2

Short Descriptions

`libunistring.so` provides the unicode string library API

libunwind-1.6.2

Introduction to libunwind

The libunwind package contains a portable and efficient C programming interface (API) to determine the call-chain of a program. The API additionally provides the means to manipulate the preserved (callee-saved) state of each call-frame and to resume execution at any point in the call-chain (non-local goto). The API supports both local (same-process) and remote (across-process) operation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.savannah.nongnu.org/releases/libunwind/libunwind-1.6.2.tar.gz>
- Download MD5 sum: f625b6a98ac1976116c71708a73dc44a
- Download size: 884 KB
- Estimated disk space required: 20 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libunwind Dependencies

Optional

[texlive-20240312](#) (for latex2man)

Installation of libunwind

Install libunwind by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`. Two tests, run-coredump-unwind and run-coredump-unwind-mdi are known to fail.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libunwind.so, libunwind-coredump.so, libunwind-generic.so, libunwind-ptrace.so, libunwind-setjmp.so, and libunwind-x86_64.so (libunwind-x86.so on i686)

Installed Directories: None

liburcu-0.14.0

Introduction to liburcu

The userspace-rcu package provides a set of userspace RCU (read-copy-update) libraries. These data synchronization libraries provide read-side access which scales linearly with the number of cores. It does so by allowing multiple copies of

a given data structure to live at the same time, and by monitoring the data structure accesses to detect grace periods after which memory reclamation is possible.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ltng.org/files/urcu/userspace-rcu-0.14.0.tar.bz2>
- Download MD5 sum: 0cd5647788b048a5d6bbdb3b08d46299
- Download size: 648 KB
- Estimated disk space required: 25 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Installation of liburcu

Install liburcu by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --docdir=/usr/share/doc/liburcu-0.14.0 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: liburcu.so, liburcu-bp.so, liburcu-cds.so, liburcu-common.so, liburcu-mb.so, liburcu-memb.so, liburcu-qsbr.so, and liburcu-signal.so

Installed Directories: /usr/include/urcu and /usr/share/doc/liburcu-0.14.0

libusb-1.0.27

Introduction to libusb

The libusb package contains a library used by some applications for USB device access.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libusb/libusb/releases/download/v1.0.27/libusb-1.0.27.tar.bz2>
- Download MD5 sum: 1fb61afe370e94f902a67e03eb39c51f
- Download size: 632 KB
- Estimated disk space required: 5.7 MB
- Estimated build time: less than 0.1 SBU

libusb Dependencies

Optional

[Doxygen-1.12.0](#)

Configuring Libusb

To access raw USB devices (those not treated as a disk by the mass-storage driver), appropriate support must be available in the kernel. Check your kernel configuration:

```
Device Drivers --->
    [*] USB support --->                                [USB_SUPPORT]
```

```

<*/M> Support for Host-side USB [USB]
[*] PCI based USB host interface [USB_PCI]
# These are most common USB controller drivers for PC-like systems.
# For modern systems often [USB_XHCI_HCD] is the only one needed
# even if the system has USB 2.0 ports:
< */M> xHCI HCD (USB 3.0) support [USB_XHCI_HCD]
< */M> EHCI HCD (USB 2.0) support [USB_EHCI_HCD]
< */M> OHCI HCD (USB 1.1) support [USB_OHCI_HCD]

```

For more details on setting up USB devices, see [the section called "USB Device Issues".](#)

Installation of libusb

Install libusb by running the following commands:

```

./configure --prefix=/usr --disable-static &&
make

```

If Doxygen is installed and you wish to build the API documentation, issue the following commands:

```

pushd doc &&
doxygen -u doxygen.cfg &&
make docs &&
popd

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you built the API documentation, install it using the following commands as the `root` user:

```

install -v -d -m755 /usr/share/doc/libusb-1.0.27/apidocs &&
install -v -m644 doc/api-1.0/* \
/usr/share/doc/libusb-1.0.27/apidocs

```

Contents

Installed Programs: None

Installed Library: libusb-1.0.so

Installed Directories: /usr/include/libusb-1.0 and /usr/share/doc/libusb-1.0.27

Short Descriptions

libusb-1.0.so contains API functions used for accessing USB hardware

libuv-1.48.0

Introduction to libuv

The libuv package is a multi-platform support library with a focus on asynchronous I/O.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://dist.libuv.org/dist/v1.48.0/libuv-v1.48.0.tar.gz>
- Download MD5 sum: a808517c32ebd07c561bf21a4e30aeab
- Download size: 1.3 MB
- Estimated disk space required: 14 MB (add 16 MB for tests)
- Estimated build time: less than 0.1 SBU (add 0.5 SBU for tests)

libuv Dependencies

Optional

[sphinx-8.0.2](#)

Installation of libuv

Install libuv by running the following commands:

Caution

The `sh autogen.sh` command below fails if the ACLOCAL environment variable is set as specified in [Xorg-7](#). If it is used, ACLOCAL needs to be unset for this package and then reset for other packages.

```
sh autogen.sh          &&
./configure --prefix=/usr --disable-static &&
make
```

If you installed the optional [sphinx-8.0.2](#) python module, create the man page:

```
make man -C docs
```

If you want to run the tests, run: `make check` as a non-root user.

Now, as the `root` user:

```
make install
```

If you built the man page, install it as the `root` user:

```
install -Dm644 docs/build/man/libuv.1 /usr/share/man/man1
```

Contents

Installed Programs: None

Installed Library: libuv.so

Installed Directory: /usr/include/uv

Short Descriptions

`libuv.so` contains API functions for asynchronous I/O operations

libxkbcommon-1.7.0

Introduction to libxkbcommon

libxkbcommon is a keymap compiler and support library which processes a reduced subset of keymaps as defined by the XKB specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xkbcommon.org/download/libxkbcommon-1.7.0.tar.xz>
- Download MD5 sum: b05b1a0d473189efb2dd995dd944f152
- Download size: 524 KB
- Estimated disk space required: 26 MB (with tests)
- Estimated build time: 0.5 SBU (with tests)

libxkbcommon Dependencies

Required

[xkeyboard-config-2.42](#) (runtime)

Recommended

[libxcb-1.17.0](#), [Wayland-1.23.0](#), and [wayland-protocols-1.36](#)

Optional

[Doxygen-1.12.0](#) (for generating the documentation) and [xvfb](#) (from [Xorg-Server-21.1.13](#) or [Xwayland-24.1.2](#))

Installation of libxkbcommon

Install libxkbcommon by running the following commands:

```
mkdir build &&
cd      build &&

meson setup ..          \
    --prefix=/usr        \
    --buildtype=release  \
    -D enable-docs=false &&
ninja
```

To test the results, ensure [xvfb](#) and [xkeyboard-config-2.42](#) are available, then issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D enable-docs=true`: This switch enables documentation generation. The [Doxygen-1.12.0](#) is required.

`mv -v /usr/share/doc/libxkbcommon,-1.7.0`: If you built the documentation, use this command to install it in a versioned directory.

Contents

Installed Programs: `xkbcli`

Installed Libraries: `libxkbcommon.so`, `libxkbcommon-x11.so`, and `libxkbregistry.so`

Installed Directories: `/usr/include/xkbcommon`, `/usr/libexec/xkbcommon`, and `/usr/share/doc/libxkbcommon-1.7.0`

Short Descriptions

<code>xkbcli</code>	provides a debugger and compiler for XKB keymaps
<code>libxkbcommon.so</code>	contains the libxkbcommon API functions
<code>libxkbcommon-x11.so</code>	contains the libxkbcommon X11 specific API functions
<code>libxkbregistry.so</code>	contains a list of available XKB models, layouts, and variants for a given ruleset

libxml2-2.13.3

Introduction to libxml2

The libxml2 package contains libraries and utilities used for parsing XML files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libxml2/2.13/libxml2-2.13.3.tar.xz>
- Download MD5 sum: aeb92dc58cea91c95a10751327193a11
- Download size: 2.5 MB
- Estimated disk space required: 111 MB (with tests)
- Estimated build time: 0.4 SBU (Using parallelism=4; with tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/libxml2-2.13.3-upstream_fix-2.patch
 - Optional Test Suite: <https://www.w3.org/XML/Test/xmlts20130923.tar.gz> - This enables `make check` to do complete testing.

libxml2 Dependencies

Recommended

ICU-75.1

Optional

Valgrind-3.23.0 (may be used in the tests)

Installation of libxml2

At first, fix an issue causing `xmlcatalog` to emit bogus warnings when a new catalog file is created. The patch will modify the building system, so regenerate the building scripts after applying the patch:

```
patch -Np1 -i ../../libxml2-2.13.3-upstream_fix-2.patch
```

Install libxml2 by running the following commands:

```
./configure --prefix=/usr           \
            --sysconfdir=/etc        \
            --disable-static         \
            --with-history          \
            --with-icu               \
PYTHON=/usr/bin/python3 \
            --docdir=/usr/share/doc/libxml2-2.13.3 &&
make
```

If you downloaded the test suite, issue the following command:

```
tar xf ../xmlts20130923.tar.gz
```

To test the results, issue: `make check > check.log`. This command will print several lines of error messages like "Failed to parse xstc/..." because some test files are missing and these messages can be safely ignored. A summary of the results can be obtained with `grep -E '^Total|expected|Ran' check.log`. If [Valgrind-3.23.0](#) is installed and you want to check for memory leaks, replace `check` with `check-valgrind`.

Note

The tests use <http://localhost/> to test parsing of external entities. If the machine where you run the tests serves as a web site, the tests may hang, depending on the content of the file served. It is therefore recommended to shut down the server during the tests, as the `root` user:

```
systemctl stop httpd.service
```

Now, as the *root* user:

make install

Finally, prevent some packages from unnecessarily linking to ICU using the following commands as the `root` user:

```
rm -vf /usr/lib/libxml2.la &&
sed '/libs=/s/xml2.*/xml2"/' -i /usr/bin/xml2-config
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

--with-history: This switch enables Readline support when running `xmlcatalog` or `xmlint` in shell mode.

`--with-icu`: This switch enables support for ICU, which provides additional Unicode support. This is needed for some packages in BLFS, such as for QtWebEngine.

`PYTHON=/usr/bin/python3`: Allows building the libxml2 module with Python3 instead of Python2.

Contents

Installed Programs: xml2-config, xmllcatalog, and xmllint

Installed Libraries: libxml2.so

Installed Directories: /usr/include/libxml2, /usr/lib/cmake/libxml2, /usr/share/doc/libxml2-2.13.3, and /usr/share/gtk-doc/html/libxml2

Short Descriptions

<code>xml2-</code> <code>config</code>	determines the compile and linker flags that should be used to compile and link programs that use libxml2
<code>xmllcatalog</code>	is used to monitor and manipulate XML and SGML catalogs
<code>xmllint</code>	parses XML files and outputs reports (based upon options) to detect errors in XML coding
<code>libxml2.so</code>	provides functions for programs to parse files that use the XML format

libxmlb-0.3.19

Introduction to libxmlb

The libxmlb package contains a library and a tool which help create and query binary XML blobs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/hughsie/libxmlb/releases/download/0.3.19/libxmlb-0.3.19.tar.xz>
- Download MD5 sum: 0749fbcd7179b800c68ed8a6aa31cb74
- Download size: 104 KB
- Estimated disk space required: 3.8 MB
- Estimated build time: 0.1 SBU

libxmlb Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended)

Optional

[GTK-Doc-1.34.0](#) and [libstemmer](#)

Installation of libxmlb

Install libxmlb by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D gtkdoc=false .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtkdoc=false`: This switch disables building the API documentation. Remove it if you have [GTK-Doc-1.34.0](#) installed and wish to build the API documentation.

`-D introspection=false`: Use this switch if you do not have built [GLib-2.80.4](#) with GObject Introspection.

`-D stemmer=true`: Use this switch if you have [libstemmer](#) installed and wish to build stemmer support.

Contents

Installed Programs: xb-tool

Installed Libraries: libxmlb.so

Installed Directories: /usr/include/libxmlb-2, /usr/libexec/installed-tests/libxmlb, and /usr/share/gtk-doc/html/libxmlb (if gtk-doc was enabled)

Short Descriptions

<code>xb-tool</code>	is a tool used to create, dump, or query a binary XML blob
<code>libxmlb.so</code>	contains functions that allow for creating and querying binary XML blobs

libxslt-1.1.42

Introduction to libxslt

The libxslt package contains XSLT libraries used for extending `libxml2` libraries to support XSLT files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libxslt/1.1/libxslt-1.1.42.tar.xz>
- Download MD5 sum: 56bc5d89aa39d62002961c150fec08a0
- Download size: 1.6 MB
- Estimated disk space required: 34 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

libxslt Dependencies

Required

[libxml2-2.13.3](#)

Recommended (at runtime)

[docbook-xml-4.5](#) and [docbook-xsl-nons-1.79.2](#)

Note

Although it is not a direct dependency, many applications using libxslt will expect [docbook-xml-4.5](#) and [docbook-xsl-nons-1.79.2](#) to be present.

Optional

[libgcrypt-1.11.0](#)

Installation of libxslt

Install libxslt by running the following commands:

```
./configure --prefix=/usr\n          \\\n          --disable-static\n          \\
```

```
--docdir=/usr/share/doc/libxslt-1.1.42 &&  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: `xslt-config` and `xsltproc`

Installed Libraries: `libexslt.so`, `libxslt.so` and optionally, `libxsltmod.so` Python module

Installed Directories: `/usr/include/libexslt`, `/usr/include/libxslt`, `/usr/lib/cmake/libxslt`, `/usr/lib/libxslt-plugins`, `/usr/share/gtk-doc/lib{xslt,exslt}`, `/usr/share/doc/libxslt-1.1.42`, and `/usr/share/doc/libxslt-python-1.1.42`

Short Descriptions

<code>xslt-config</code>	is used to find out the pre-processor, linking and compiling flags necessary to use the <code>libxslt</code> libraries in 3rd-party programs
<code>xsltproc</code>	is used to apply XSLT stylesheets to XML documents
<code>libexslt.so</code>	is used to provide extensions to XSLT functions
<code>libxslt.so</code>	provides extensions to the <code>libxml2</code> libraries to parse files that use the XSLT format

libwacom-2.12.2

Introduction to libwacom

The `libwacom` package contains a library used to identify graphics tablets from Wacom or various other vendors and their model-specific features.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/linuxwacom/libwacom/releases/download/libwacom-2.12.2/libwacom-2.12.2.tar.xz>
- Download MD5 sum: 7005d961cc44995d8c04d586fa060c63
- Download size: 152 KB
- Estimated disk space required: 7.9 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

libwacom Dependencies

Required

[libevdev-1.13.2](#) and [libgudev-238](#)

Recommended

[libxml2-2.13.3](#)

Optional

[Doxygen-1.12.0](#), [git-2.46.0](#), [librsvg-2.58.3](#), [Valgrind-3.23.0](#) (optional for some tests), and [pytest-8.3.2](#) with [python-libevdev](#) and [pyudev](#)

Installation of libwacom

Install libwacom by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D tests=disabled &&
ninja
```

To test the results, issue: `ninja test`. To run additional tests, install [pytest-8.3.2](#), python-libevdev, and pyudev, then remove the "-D tests=disabled" option from the meson line above.

If upgrading from a previous version of libwacom, remove the old device database installation to prevent a potential duplicated match of devices in case some old database files are not overwritten:

```
rm -rf /usr/share/libwacom
```

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D tests=disabled`: This parameter disables some of the more advanced tests because they require [pytest-8.3.2](#) and other two Python modules beyond the scope of BLFS to work properly.

Contents

Installed Programs: libwacom-list-devices, libwacom-list-local-devices, libwacom-show-stylus, and libwacom-update-db

Installed Libraries: libwacom.so

Installed Directories: /usr/include/libwacom-1.0 and /usr/share/libwacom

Short Descriptions

<code>libwacom-list-devices</code>	lists all tablet devices that are supported by libwacom
<code>libwacom-list-local-devices</code>	lists tablet devices that are connected to the system
<code>libwacom-show-stylus</code>	lists tablet stylus IDs
<code>libwacom-update-db</code>	updates the system according to the current set of tablet data files
<code>libwacom.so</code>	contains functions used for accessing Wacom information

libyaml-0.2.5

Introduction to libyaml

The yaml package contains a C library for parsing and emitting YAML (YAML Ain't Markup Language) code.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/yaml/libyaml/releases/download/0.2.5/yaml-0.2.5.tar.gz>
- Download MD5 sum: bb15429d8fb787e7d3f1c83ae129a999
- Download size: 596 KB
- Estimated disk space required: 6.4 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

libyaml Dependencies

Optional

Installation of libyaml

Install libyaml by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libyaml.so

Installed Directories: None

Short Descriptions

libyaml.so contains API functions for parsing and emitting YAML code

log4cplus-2.1.1

Introduction to log4cplus

log4cplus is an easy to use C++20 logging API providing thread-safe, flexible, and arbitrarily granular control over log management and configuration. It is modeled after the Java log4j API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/log4cplus/log4cplus/releases/download/REL_2_1_1/log4cplus-2.1.1.tar.xz
- Download MD5 sum: 6ee2555be39cd269086cc871c834e43f
- Download size: 919 KB
- Estimated disk space required: 73 MB
- Estimated build time: 0.3 SBU (Using parallelism=4; add 0.5 SBU for tests)

log4cplus Dependencies

Optional

[qt5-components-5.15.14](#)

Installation of log4cplus

Build log4cplus by running the following command:

```
./configure --prefix=/usr &&  
make
```

To test the results, issue: `make check`.

Now, install the package as the `root` user:

make install

Command Explanations

--with-qt5: This option enables compilation of a separate shared library (liblog4cplusqt5debugappender) that implements Qt5DebugAppender. It requires Qt5 and pkg-config to be available.

Contents

Installed Programs: None

Installed Library: log4cplus.so

Installed Directories: None

Short Descriptions

`log4cplus.so` is the `log4cplus` library

LZO-2.10

Introduction to LZO

LZO is a data compression library which is suitable for data decompression and compression in real-time. This means it favors speed over compression ratio.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.oberhummer.com/opensource/lzo/download/lzo-2.10.tar.gz>
 - Download MD5 sum: 39d3f3f9c55c87b1e5d6888e1420f4b5
 - Download size: 588 KB
 - Estimated disk space required: 12 MB
 - Estimated build time: 0.3 SBU (with tests)

Installation of LZO

Install LZO by running the following commands:

```
./configure --prefix=/usr  
          --enable-shared  
          --disable-static  
          --docdir=/usr/share/doc/lzo-2.10 &&  
  
make
```

To test the results, issue: `make check`. All the checks should pass. Now issue `make test` to run the full suite of tests.

Now, as the *root* user:

make install

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries:

Installed Directories: /usr/include/lzo and /usr/share/doc/lzo

Short Descriptions

`liblzo2.so` is a data compression and decompression library

mtdev-1.1.7

Introduction to mtdev

The mtdev package contains Multitouch Protocol Translation Library which is used to transform all variants of kernel MT (Multitouch) events to the slotted type B protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://bitmath.org/code/mtdev/mtdev-1.1.7.tar.bz2>
- Download MD5 sum: 483ed7fdf7c1e7b7375c05a62848cce7
- Download size: 296 KB
- Estimated disk space required: 2.5 MB
- Estimated build time: less than 0.1 SBU

Installation of mtdev

Install mtdev by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: `mtdev-test`

Installed Library: `libmtdev.so`

Installed Directories: None

Short Descriptions

`mtdev-test` is a tool to test `libmtdev`
`libmtdev.so` contains Multitouch Protocol Translation API functions

Node.js-20.16.0

Introduction to Node.js

Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://nodejs.org/dist/v20.16.0/node-v20.16.0.tar.xz>
- Download MD5 sum: 7832164e52741a743106c5aadd34cdee
- Download size: 40 MB
- Estimated disk space required: 1.0 GB (add 67 MB for tests)

- Estimated build time: 12 SBU (add 2.9 SBU for tests: both using parallelism=8 and 8 CPUs online, parts of the tests will use all online CPUs)

Node.js Dependencies

Required

Which-2.21

Recommended

[Brotli-1.1.0](#), [c-ares-1.33.0](#), [ICU-75.1](#), [libuv-1.48.0](#), and [nghttp2-1.62.1](#)

Optional

[http-parser](#) and [npm](#) (an internal copy of `npm` will be installed if not present)

Note

An Internet connection is needed for some tests of this package.

Installation of Node.js

Build Node.js by running the following commands:

```
./configure --prefix=/usr \
            --shared-brotli \
            --shared-cares \
            --shared-libuv \
            --shared-openssl \
            --shared-nghttp2 \
            --shared-zlib \
            --with-intl=system-icu &
```

To test the results, issue: `make test-only`. Out of 4096 tests, 25 in the 'parallel' test suite are known to fail.

Now, as the *root* user:

```
make install &&
ln -sf node /usr/share/doc/node-20.16.0
```

Command Explanations

--with-intl=system-icu: use the system version of icu. Other values are full-icu (to build a local, full icu library) and small-icu (to build a local, minimal icu library).

--shared-{*brotli, cares, libuv, nghttp2, openssl, zlib*}: use the system installed libraries instead of local copies.

--without-npm: do not build npm (use if you'd like to build a separate npm later).

--shared-**http-parser**: use the system installed library instead of a local copy.

Contents

Installed Programs: corepack, node, npm, and npx

Installed Library: None

Installed Directories: /usr/include/node, /usr/lib/node_modules/{corepack,npm}, /usr/share/doc/{node,node-20.16.0}, and /usr/share/systemtap/tapset

Short Descriptions

<code>corepack</code>	is an experimental tool to help with managing versions of package managers.
<code>node</code>	is the server-side JavaScript runtime
<code>npm</code>	is the Node.js package manager

/usr/lib/node_modules/npm/ is the installation root for Node.js executables and libraries

npth-1.7

Introduction to NPth

The NPth package contains a very portable POSIX/ANSI-C based library for Unix platforms which provides non-preemptive priority-based scheduling for multiple threads of execution (multithreading) inside event-driven applications. All threads run in the same address space of the server application, but each thread has its own individual program-counter, run-time stack, signal mask and errno variable.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/npth/npth-1.7.tar.bz2>
- Download MD5 sum: 286274d0106ec408efebe4a399975b11
- Download size: 308 KB
- Estimated disk space required: 2.6 MB (with checks)
- Estimated build time: less than 0.1 SBU (with checks)

Installation of NPth

Install NPth by running the following commands:

```
./configure --prefix=/usr &&  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: npth-config

Installed Library: libnpth.so

Installed Directory: None

Short Descriptions

`npth-config` is a utility used to configure and build applications based on the npth library. It can be used to query the C compiler and linker flags which are required to correctly compile and link the application against the npth library

`libnpth.so` contains the API functions used by the New Portable Threads Library

NSPR-4.35

Introduction to NSPR

Netscape Portable Runtime (NSPR) provides a platform-neutral API for system level and libc like functions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.mozilla.org/pub/nspr/releases/v4.35/src/nspr-4.35.tar.gz>
- Download MD5 sum: 5e0acf9fbdde85181bdd510f4624841
- Download size: 1.0 MB
- Estimated disk space required: 9.6 MB

- Estimated build time: less than 0.1 SBU

Installation of NSPR

Install NSPR by running the following commands:

```
cd nspr &&
sed -i '/^RELEASE/s/^|#/! pr/src/misc/Makefile.in &&
sed -i 's|${LIBRARY} ||' config/rules.mk &&
./configure --prefix=/usr \
--with-mozilla \
--with-pthreads \
$([ $(uname -m) = x86_64 ] && echo --enable-64bit) &&
make
```

The test suite is designed for testing changes to nss or nspr and is not particularly useful for checking a released version (e.g. it needs to be run on a non-optimized build with both nss and nspr directories existing alongside each other). For further details, see the Editor Notes for nss at <https://wiki.linuxfromscratch.org/blfs/wiki/nss>

Now, as the `root` user:

```
make install
```

Command Explanations

`sed -ri '/^RELEASE/s/^|#/! pr/src/misc/Makefile.in`: This sed disables installing two unneeded scripts.

`sed -i 's#${LIBRARY} ##' config/rules.mk`: This sed disables installing the static libraries.

`--with-mozilla`: This parameter adds Mozilla support to the libraries (required if you want to build any other Mozilla products and link them to these libraries).

`--with-pthreads`: This parameter forces use of the system pthread library.

`--enable-64bit`: The `--enable-64bit` parameter is *required* on an x86_64 system to prevent `configure` failing with a claim that this is a system without pthread support. The `[$(uname -m) = x86_64]` test ensures it has no effect on a 32 bit system.

Contents

Installed Programs: nspr-config

Installed Libraries: libnspr4.so, libplc4.so, and libplds4.so

Installed Directories: /usr/include/nspr

Short Descriptions

<code>nspr-config</code>	provides compiler and linker options to other packages that use NSPR
<code>libnspr4.so</code>	contains functions that provide platform independence for non-GUI operating system facilities such as threads, thread synchronization, normal file and network I/O, interval timing and calendar time, basic memory management and shared library linking
<code>libplc4.so</code>	contains functions that implement many of the features offered by libnspr4
<code>libplds4.so</code>	contains functions that provide data structures

PCRE2-10.44

Introduction to PCRE2

The PCRE2 package contains a new generation of the Perl Compatible Regular Expression libraries. These are useful for implementing regular expression pattern matching using the same syntax and semantics as Perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/PCRE2Project/pcre2/releases/download/pcre2-10.44/pcre2-10.44.tar.bz2>

- Download MD5 sum: 9d1fe11e2e919c7b395e3e8f0a5c3eec
- Download size: 1.8 MB
- Estimated disk space required: 20 MB (with tests)
- Estimated build time: 0.5 SBU (with tests)

PCRE2 Dependencies

Optional

[Valgrind-3.23.0](#) and [libedit](#)

Installation of PCRE2

Install PCRE2 by running the following commands:

```
./configure --prefix=/usr \
            --docdir=/usr/share/doc/pcre2-10.44 \
            --enable-unicode \
            --enable-jit \
            --enable-pcre2-16 \
            --enable-pcre2-32 \
            --enable-pcre2grep-libz \
            --enable-pcre2grep-libbz2 \
            --enable-pcre2test-libreadline \
            --disable-static
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-unicode`: This switch enables Unicode support and includes the functions for handling UTF-8/16/32 character strings in the library.

`--enable-pcre2-16`: This switch enables 16 bit character support.

`--enable-pcre2-32`: This switch enables 32 bit character support.

`--enable-pcre2grep-libz`: This switch adds support for reading .gz compressed files to pcre2grep.

`--enable-pcre2grep-libbz2`: This switch adds support for reading .bz2 compressed files to pcre2grep.

`--enable-pcre2test-libreadline`: This switch adds line editing and history features to the pcre2test program.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-jit`: this option enables Just-in-time compiling, which can greatly speed up pattern matching.

Contents

Installed Programs: pcre2-config, pcre2grep, and pcre2test.

Installed Libraries: libpcre2-8.so, libpcre2-16.so, libpcre2-32.so, and libpcre2-posix.so

Installed Directory: /usr/share/doc/pcre2-10.44

Short Descriptions

<code>pcre2grep</code>	is a version of grep that understands Perl compatible regular expressions.
<code>pcre2test</code>	can test a Perl compatible regular expression.
<code>pcre2-config</code>	outputs compilation information to programs linking against the PCRE2 libraries

Introduction to Popt

The popt package contains the popt libraries which are used by some programs to parse command-line options.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://ftp.rpm.org/popt/releases/popt-1.x/popt-1.19.tar.gz>
- Download MD5 sum: eaa2135fddb6eb03f2c87ee1823e5a78
- Download size: 584 KB
- Estimated disk space required: 6.9 MB (includes installing documentation and tests)
- Estimated build time: less than 0.1 SBU (with tests)

popt Dependencies

Optional

[Doxygen-1.12.0](#) (for generating documentation)

Installation of Popt

Install popt by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

If you have [Doxygen-1.12.0](#) installed and wish to build the API documentation, issue:

```
sed -i 's@./@src/@' Doxyfile &&  
doxygen
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you built the API documentation, install it using the following commands issued by the `root` user:

```
install -v -m755 -d /usr/share/doc/popt-1.19 &&  
install -v -m644 doxygen/html/* /usr/share/doc/popt-1.19
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libpopt.so

Installed Directories: /usr/share/doc/popt-1.19

Short Descriptions

libpopt.so is used to parse command-line options

Protobuf-27.3

Introduction to Protobuf

The Protobuf package contains utilities and libraries for using data in Google's data interchange format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/protocolbuffers/protobuf/releases/download/v27.3/protobuf-27.3.tar.gz>
- Download MD5 sum: c7feee9c4a6856d39e2b9d6876a487f5
- Download size: 6.0 MB
- Estimated disk space required: 79 MB
- Estimated build time: 1.6 SBU (with parallelism=4)

Protobuf Dependencies

Required

[Abseil-cpp-20240722.0](#) and [CMake-3.30.2](#)

Optional

[gtest \(for tests\)](#).

Installation of Protobuf

Install Protobuf by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
      -D CMAKE_BUILD_TYPE=Release       \
      -D CMAKE_SKIP_INSTALL_RPATH=ON    \
      -D protobuf_BUILD_TESTS=OFF      \
      -D protobuf_ABSL_PROVIDER=package \
      -D protobuf_BUILD_LIBUPB=OFF     \
      -D protobuf_BUILD_SHARED_LIBS=ON  \
      -D utf8_range_ENABLE_INSTALL=OFF  \
      -G Ninja ..                      &&
ninja
```

This package does come with a test suite, but it requires [gtest](#), which is not part of BLFS.

Now, as the `root` user:

```
sed 's/utf8_range//' -i *.pc &&
ninja install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`sed 's/utf8_range//' -i *.pc`: This command removes a leftover reference to libutf8_range from the pkg-config files installed by this package.

`-D protobuf_BUILD_TESTS=OFF`: This parameter prevents the tests from being built because [gtest](#) is not part of BLFS.

`-D protobuf_ABSL_PROVIDER=package`: This parameter allows the build system to use the system-installed copy of [Abseil-cpp-20240722.0](#).

`-D protobuf_BUILD_SHARED_LIBS=ON`: This parameter enables building shared versions of the libraries provided by this package instead of static versions.

`-D utf8_range_ENABLE_INSTALL=OFF`: This parameter disables installing the utf8_range static library. The functions provided by this library and used by protobuf is already embedded into `libprotobuf.so`, so installing a full copy of the static library is just wasting the disk space.

Contents

Installed Programs: `protoc-26.1.0` and `protoc` (a symlink to `protoc-26.1.0`)

Installed Libraries: libprotobuf.so, libprotobuf-lite.so, and libprotoc.so

Installed Directories: /usr/include/google, /usr/include/java, /usr/include/upb_generator, and /usr/lib/cmake/protobuf

Short Descriptions

protoc	parses protocol buffer files and generates output for several programming languages and formats
libprotobuf.so	contains functions for utilizing data in Google's data interchange format
libprotobuf-lite.so	contains a simpler version of the functions for utilizing data in Google's data interchange format
libprotoc.so	contains functions used by protoc at runtime for outputting data for several programming languages and formats

Protobuf-c-1.5.0

Introduction to Protobuf-c

The Protobuf-c package contains an implementation of the Google Protocol Buffers data serialization format in C.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/protobuf-c/protobuf-c/releases/download/v1.5.0/protobuf-c-1.5.0.tar.gz>
- Download MD5 sum: 03030a3994b9f89ef166336048992484
- Download size: 496 KB
- Estimated disk space required: 60 MB
- Estimated build time: 0.6 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/protobuf-c-1.5.0-consolidated_fixes-1.patch

Protobuf-c Dependencies

Required

[Protobuf-27.3](#)

Optional

[Doxygen-1.12.0](#)

Installation of Protobuf-c

First, apply some upstream fixes to make this package compatible with protobuf-26.1:

```
patch -Np1 -i ../../protobuf-c-1.5.0-consolidated_fixes-1.patch
```

Install Protobuf-c by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: protoc-gen-c and protoc-c (symlinked to protoc-gen-c)

Installed Libraries: libprotobuf-c.so

Installed Directories: /usr/include/google and /usr/include/protobuf-c

Short Descriptions

protoc-c	generates C descriptor code from a .proto file
libprotobuf-c.so	contains a C implementation of the Google Protocol Buffers data serialization format

Qca-2.3.9

Introduction to Qca

Qca aims to provide a straightforward and cross-platform crypto API, using Qt datatypes and conventions. Qca separates the API from the implementation, using plugins known as Providers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/qca/2.3.9/qca-2.3.9.tar.xz>
- Download MD5 sum: d8aaa46356a322464f65b04d00d2bac6
- Download size: 748 KB
- Estimated disk space required: 57 MB (with tests)
- Estimated build time: 1.0 SBU (using parallelism=4; with tests)

Qca Dependencies

Required

[make-ca-1.14](#), [CMake-3.30.2](#), [Qt-6.7.2](#), and [Which-2.21](#)

Optional

[Cyrus SASL-2.1.28](#), [GnuPG-2.4.5](#), [libgcrypt-1.11.0](#), [libgpg-error-1.50](#), [nss-3.103](#), [NSPR-4.35](#), [p11-kit-0.25.5](#), [Doxygen-1.12.0](#), and [Botan](#)

Installation of Qca

Fix the location of the CA certificates:

```
sed -i 's@cert.pem@certs/ca-bundle.crt@' CMakeLists.txt
```

Install Qca by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$QT6DIR \
      -D CMAKE_BUILD_TYPE=Release \
      -D QT6=ON \
      -D QCA_INSTALL_IN_QT_PREFIX=ON \
      -D QCA_MAN_INSTALL_DIR:PATH=/usr/share/man \
      .. &&
make
```

To test the results, issue `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply a higher level of compiler optimizations.

`-D QT6=ON`: Ensure the packages is built with [Qt-6.7.2](#).

`-D QCA_MAN_INSTALL_DIR:PATH=/usr/share/man`: Install the qca man page in the normal location.

Contents

Installed Programs: mozcerts-qt6 and qcatool-qt6

Installed Libraries: libqca-qt6.so, libqca-cyrus-sasl.so, libqca-gcrypt.so, libqca-gnupg.so, libqca-logger.so, libqca-nss.so, libqca-openssl.so, and libqca-softstore.so

Installed Directories: \$QT6DIR/include/Qca-qt6, \$QT6DIR/lib/cmake/Qca-qt6, and \$QT6DIR/lib/qca-qt6

Short Descriptions

<code>mozcerts-qt6</code>	is a command line tool for converting certdata.txt into outfile.pem files
<code>qcatool-qt6</code>	is a command line tool for performing various cryptographic operations with Qca
<code>libqca-qt6.so</code>	is the Qt Cryptography Architecture (Qca) library

qcoro-0.10.0

Introduction to qcoro

This package provides a set of tools to make use of C++20 coroutines with Qt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/danvratil/qcoro/archive/v0.10.0/qcoro-0.10.0.tar.gz>
- Download MD5 sum: 2af4e86cd77493cd41ba5ffcac33d5f4
- Download size: 148 KB
- Estimated disk space required: 8.4 MB (Add 19 MB for tests)
- Estimated build time: 0.4 SBU (Add 0.5 SBU for tests)

qcoro Dependencies

Required

[Qt-6.7.2](#)

Installation of qcoro

Install qcoro by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$QT6DIR \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-D QCORO_BUILD_EXAMPLES=OFF \
-D BUILD_SHARED_LIBS=ON \
...
&&
make
```

To test this package, remove the 'BUILD_TESTING=OFF' parameter above and run:

```
make test
```

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply a higher level of compiler optimizations.

Contents

Installed Programs: None

Installed Libraries: libQCoro6Core.so, libQCoro6DBus.so, libQCoro6Network.so, libQCoro6Qml.so, libQCoro6Quick.so, and libQCoro6WebSockets.so

Installed Directories: Nine directories in `$QT6DIR/lib/cmake/` and `$QT6DIR/include/qcoro6`

SpiderMonkey from firefox-115.14.0

Introduction to SpiderMonkey

SpiderMonkey is Mozilla's JavaScript and WebAssembly Engine, written in C++ and Rust. In BLFS, the source code of SpiderMonkey is taken from Firefox.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.mozilla.org/pub/firefox/releases/115.14.0esr/source/firefox-115.14.0esr.source.tar.xz>
- Download MD5 sum: deb750363b35d86629f824c0f8ba7f56
- Download size: 484 MB
- Estimated disk space required: 3.6 GB (40 MB installed after removing 36 MB static lib; add 34 MB for the main tests and 37 MB for the jit tests)
- Estimated build time: 1.9 SBU (with parallelism=4; add 1.1 SBU for the main tests and 3.7 SBU for the jit tests)

SpiderMonkey Dependencies

Required

[ICU-75.1](#), [rustc-1.80.1](#), and [Which-2.21](#)

Recommended

[LLVM-18.1.7](#) (with Clang, required for 32-bit systems without SSE2 capabilities)

Important

If you are building this package on a 32-bit system, and Clang is not installed or you're overriding the default compiler choice with the environment variable `CXX`, please read the Command Explanations section first.

Installation of SpiderMonkey

Note

Unlike most other packages in BLFS, the instructions below require you to untar `firefox-115.14.0esr.tar.xz` and change into the `firefox-115.14.0` directory.

Extracting the tarball will reset the permissions of the current directory to 0755 if you have permission to do that. If you do this in a directory where the sticky bit is set, such as `/tmp` it will end with error messages:

```
tar: ..: Cannot utime: Operation not permitted
tar: ..: Cannot change mode to rwxr-xr-t: Operation not permitted
```

```
tar: Exiting with failure status due to previous errors
```

This does finish with non-zero status, but it does *NOT* mean there is a real problem. Do not untar as the `root` user in a directory where the sticky bit is set - that will unset it.

Install SpiderMonkey by running the following commands:

Note

If you are compiling this package in chroot you must ensure that `/dev/shm` is mounted. If you do not do this, the Python configuration will fail with a traceback report referencing `/usr/lib/pythonN.N/multiprocessing/synchronize.py`. As the `root` user, run:

```
mountpoint -q /dev/shm || mount -t tmpfs devshm /dev/shm
```

Compiling the C++ code respects `$MAKEFLAGS` and defaults to 'j1', the rust code will use all processors.

```
mkdir obj &&
cd obj &&

./js/src/configure --prefix=/usr \
    --disable-debug-symbols \
    --disable-jemalloc \
    --enable-readline \
    --with-intl-api \
    --with-system-icu \
    --with-system-zlib &&
make
```

To run the SpiderMonkey test suite, issue: `make -C js/src check-jstests JSTESTS_EXTRA_ARGS="--timeout 300 --wpt=disabled"`. It's recommended to redirect the output into a log. Because we are building with system ICU, 39 tests (out of a total of more than 50,000) are known to fail. The test suite is executed with all CPU cores available: even in a cgroup with less cores assigned, it still attempts to spawn as many testing jobs as the number of *all* cores in the system; fortunately the kernel still won't run these jobs on cores not assigned to the cgroup so the CPU usage is still controlled.

To run the JIT test suite, issue: `make -C js/src check-jit-test JITTEST_EXTRA_ARGS="--timeout 300"`. Like the SpiderMonkey test suite, the number of test jobs is same as the number of all CPU cores in the system even if a cgroup is used. To make things worse, there are six tests which will use 3 GB each of system memory, so the peak memory usage may be up to 18 GB if the number of cores is six or more. Running the JIT test suite without enough memory may invoke the kernel OOM killer and cause stability issues. If you don't have enough system memory available, append `-jN` after `--timeout 300` with N replaced by the number of parallel test jobs you want to start. For example, if you have 16 GB system memory available and 8 CPU cores, issue `make -C js/src check-jit-test JITTEST_EXTRA_ARGS="--timeout=300 -j5"` to run the test with 5 parallel jobs so the memory usage won't exceed 15 GB.

Caution

An issue in the installation process causes any running program which links to SpiderMonkey shared library (for example, GNOME Shell) to crash if SpiderMonkey is reinstalled, or upgraded or downgraded without a change of the major version number (115 in 115.14.0). To work around this issue, remove the old version of the SpiderMonkey shared library before installation:

```
rm -fv /usr/lib/libmozjs-115.so
```

Now, as the `root` user:

```
make install &&
rm -v /usr/lib/libjs_static.ajs &&
sed -i '/@NSPR_CFLAGS@/d' /usr/bin/js115-config
```

Command Explanations

`--disable-debug-symbols`: Don't generate debug symbols since they are very large and most users won't need it. Remove it if you want to debug SpiderMonkey.

`--disable-jemalloc`: This switch disables the internal memory allocator used in SpiderMonkey. jemalloc is only intended for the Firefox browser environment. For other applications using SpiderMonkey, the application may crash as items allocated in the jemalloc allocator are freed on the system (glibc) allocator.

`--enable-readline`: This switch enables Readline support in the SpiderMonkey command line interface.

`--with-intl-api`: This enables the internationalization functions required by Gjs.

`--with-system-*`: These parameters allow the build system to use system versions of the above libraries. These are required for stability.

`rm -v /usr/lib/libjs_static.ajs`: Remove a large static library which is not used by any BLFS package.

`sed -i '/@NSPR_CFLAGS@/d' /usr/bin/js115-config`: Prevent `js115-config` from using buggy CFLAGS.

`CC=gcc CXX=g++`: BLFS used to prefer to use gcc and g++ instead of upstream's defaults of the clang programs. With the release of gcc-12 the build takes longer with gcc and g++, primarily because of extra warnings, and is bigger. Pass these environment variables to the configure script if you wish to continue to use gcc, g++ (by exporting them and unset them after the installation, or simply prepending them before the `../js/src/configure` command). If you are building on a 32-bit system, also see below.

`CXXFLAGS="-msse2 -mfpmath=sse"`: Use SSE2 instead of 387 for double-precision floating-point operations. It's needed by GCC to satisfy the expectations of upstream (Mozilla) developers with floating-point arithmetic. Use it if you are building this package on a 32-bit system with GCC (if Clang is not installed or GCC is explicitly specified). Note that this will cause SpiderMonkey to crash on a processor without SSE2 capability. If you are running the system on such an old processor, Clang is strictly needed. This setting is not needed on 64-bit systems because all 64-bit x86 processors support SSE2 and the 64-bit compilers (both Clang and GCC) use SSE2 by default.

Contents

Installed Programs: `js115` and `js115-config`

Installed Libraries: `libmozjs-115.so`

Installed Directories: `/usr/include/mozjs-115`

Short Descriptions

<code>js115</code>	provides a command line interface to the JavaScript engine
<code>js115-config</code>	is used to find the SpiderMonkey compiler and linker flags
<code>libmozjs-115.so</code>	contains the Mozilla JavaScript API functions

SPIRV-Headers-1.3.290.0

Introduction to SPIRV-Headers

The SPIRV-Headers package contains headers that allow for applications to use the SPIR-V language and instruction set with Vulkan. SPIR-V is a binary intermediate language for representing graphical shader stages and compute kernels for multiple Khronos APIs, including OpenGL and Vulkan.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KhronosGroup/SPIRV-Headers/archive/vulkan-sdk-1.3.290.0/SPIRV-Headers-1.3.290.0.tar.gz>
- Download MD5 sum: e9da8c949d89084b8a0a6b128ca6a30d
- Download size: 512 KB
- Estimated disk space required: 4.4 MB
- Estimated build time: less than 0.1 SBU

SPIRV-Headers Dependencies

Required

[CMake-3.30.2](#)

Installation of SPIRV-Headers

Note

This tarball SPIRV-Headers-1.3.290.0.tar.gz will extract to the directory SPIRV-Headers-vulkan-sdk-1.3.290.0.

Install SPIRV-Headers by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr -G Ninja .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/include/spirv and /usr/share/cmake/SPIRV-Headers

SPIRV-Tools-1.3.290.0

Introduction to SPIRV-Tools

The SPIRV-Tools package contains libraries and utilities for processing SPIR-V modules.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KhronosGroup/SPIRV-Tools/archive/vulkan-sdk-1.3.290.0/SPIRV-Tools-1.3.290.0.tar.gz>
- Download MD5 sum: 4a3f873a07f33cffd70890e17a34fe8b
- Download size: 3.0 MB
- Estimated disk space required: 59 MB
- Estimated build time: 1.0 SBU (with tests; both using parallelism=8)

SPIRV-Tools Dependencies

Required

[CMake-3.30.2](#) and [SPIRV-Headers-1.3.290.0](#)

Installation of SPIRV-Tools

Note

This tarball SPIRV-Tools-1.3.290.0.tar.gz will extract to the directory SPIRV-Tools-vulkan-sdk-1.3.290.0.

Install SPIRV-Tools by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_BUILD_TYPE=Release           \
-D SPIRV_WERROR=OFF                 \
-D BUILD_SHARED_LIBS=ON              \
-D SPIRV_TOOLS_BUILD_STATIC=OFF     \
-D SPIRV-Headers_SOURCE_DIR=/usr \
```

```
-G Ninja ... &&  
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D SPIRV_WERROR=OFF`: This switch stops the build system from treating warnings as errors.

`-D BUILD_SHARED_LIBS=ON`: This switch forces the build system to install shared libraries instead of static libraries.

`-D SPIRV_TOOLS_BUILD_STATIC=OFF`: This switch disables building static versions of the libraries.

`-D SPIRV-Headers_SOURCE_DIR`: This switch tells the build system that [SPIRV-Headers-1.3.290.0](#) is installed in `/usr`. This is needed since the build system tries to use an internal copy by default.

Contents

Installed Programs: spirv-as, spirv-cfg, spirv-dis, spirv-lesspipe.sh, spirv-link, spirv-lint, spirv-objdump, spirv-opt, spirv-reduce, and spirv-val

Installed Libraries: libSPIRV-Tools-diff.so, libSPIRV-Tools-link.so, libSPIRV-Tools-lint.so, libSPIRV-Tools-opt.so, libSPIRV-Tools-reduce.so, libSPIRV-Tools-shared.so, and libSPIRV-Tools.so

Installed Directories: /usr/include/spirv-tools and /usr/lib/cmake/SPIRV-Tools

Short Descriptions

<code>spirv-as</code>	creates a SPIR-V binary module from SPIR-V assembly text
<code>spirv-cfg</code>	shows the control flow graph in "dot" format
<code>spirv-dis</code>	disassembles a SPIR-V binary module
<code>spirv-lesspipe.sh</code>	automatically disassembles a .SPV file for 'less'
<code>spirv-link</code>	links SPIR-V binary files together
<code>spirv-lint</code>	checks a SPIR-V binary module for errors
<code>spirv-objdump</code>	dumps information from a SPIR-V binary
<code>spirv-opt</code>	performs optimizations on SPIR-V binary files
<code>spirv-reduce</code>	reduces a SPIR-V binary file
<code>spirv-val</code>	validates a SPIR-V binary file
<code>libSPIRV-Tools.so</code>	contains functions for processing SPIR-V modules

SPIRV-LLVM-Translator-18.1.3

Introduction to SPIRV-LLVM-Translator

The SPIRV-LLVM-Translator package contains a library and utility for converting between LLVM IR and SPIR-V code. This package currently only supports the OpenCL/Compute version of SPIR-V.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KhronosGroup/SPIRV-LLVM-Translator/archive/v18.1.3/SPIRV-LLVM-Translator-18.1.3.tar.gz>
- Download MD5 sum: 094ff60909ada1e76e87da90df053be6
- Download size: 1.6 MB
- Estimated disk space required: 36 MB
- Estimated build time: 0.6 SBU (with parallelism=4)

SPIRV-LLVM-Translator Dependencies

Required

Installation of SPIRV-LLVM-Translator

Install SPIRV-LLVM-Translator by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_SHARED_LIBS=ON \
      -D CMAKE_SKIP_INSTALL_RPATH=ON \
      -D LLVM_EXTERNAL_SPIRV_HEADERS_SOURCE_DIR=/usr \
      -G Ninja .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D BUILD_SHARED_LIBS=ON`: This parameter forces building shared versions of the libraries.

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D LLVM_EXTERNAL_SPIRV_HEADERS_SOURCE_DIR=/usr`: This parameter allows the build system to use the version of SPIRV-Headers that should have been installed as a dependency of SPIRV-Tools, instead of redownloading an unnecessary copy of the headers.

Contents

Installed Programs: `llvm-spirv`

Installed Libraries: `libLLVMSPIRVLib.so`

Installed Directories: `/usr/include/LLVMSPIRVLib`

Short Descriptions

<code>llvm-spirv</code>	converts between LLVM IR and SPIR-V code
<code>libLLVMSPIRVLib.so</code>	contains functions that convert between LLVM IR and SPIR-V code

Talloc-2.4.2

Introduction to Talloc

Talloc provides a hierarchical, reference counted memory pool system with destructors. It is the core memory allocator used in Samba.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.samba.org/ftp/talloc/talloc-2.4.2.tar.gz>
- Download MD5 sum: d29bdf75ce181e148c46b16a83de6d76
- Download size: 664 KB
- Estimated disk space required: 10 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

Talloc Dependencies

Optional

[docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#) and [libxslt-1.1.42](#) (To generate man pages), [GDB-15.1](#), [git-2.46.0](#), [libnsl-2.0.1](#), [libtirpc-1.3.5](#), [Valgrind-3.23.0](#), and [xfsprogs-6.9.0](#)

Installation of Talloc

Install Talloc by running the following commands:

```
./configure --prefix=/usr &&
make
```

To check the results, issue `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libpytalloc-util.cpython-312-<arch>-linux-gnu.so, libtalloc.so, and talloc.cpython-312-<arch>-linux-gnu.so (Python 3 Module)

Installed Directories: None

Short Descriptions

`libtalloc.so` contains a replacement for the Glibc malloc function

Uchardet-0.0.8

Introduction to Uchardet

The Uchardet package contains an encoding detector library which takes a sequence of bytes in an unknown character encoding and attempts to determine the encoding of the text.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/uchardet/releases/uchardet-0.0.8.tar.xz>
- Download MD5 sum: 9e267be7aee81417e5875086dd9d44fd
- Download size: 217 KB
- Estimated disk space required: 4.6 MB (with test)
- Estimated build time: less than 0.1 SBU (with test)

Uchardet Dependencies

Required

[CMake-3.30.2](#)

Installation of Uchardet

Install Uchardet by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D BUILD_STATIC=OFF \
-W no-dev .. &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D BUILD_STATIC=OFF`: This switch disables building the static version of the library.

Contents

Installed Programs: uchardet

Installed Libraries: libuchardet.so

Installed Directories: /usr/include/uchardet

Short Descriptions

<code>uchardet</code>	detects what character set is used inside of a file
<code>libuchardet.so</code>	provides an API for detecting the encoding of text in a file

Umockdev-0.18.3

Introduction to Umockdev

The Umockdev package contains a framework that allows a developer to mock devices for use in unit testing.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/martinpitt/umockdev/releases/download/0.18.3/umockdev-0.18.3.tar.xz>
- Download MD5 sum: a9b19aca965297d8e0e3b557de718856
- Download size: 484 KB
- Estimated disk space required: 6.8 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Umockdev Dependencies

Required

[libgudev-238](#), [libpcap-1.10.4](#), and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#) and [libgphoto2](#) (optional for tests)

Installation of Umockdev

Install Umockdev by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue as the `root` user: `ninja test`. One test needs to be run in an X session.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D gtk_doc=true`: Use this switch if you have [GTK-Doc-1.34.0](#) installed and wish to rebuild and install the API documentation.

Contents

Installed Programs: umockdev-record, umockdev-run, and umockdev-wrapper

Installed Libraries: libumockdev-preload.so and libumockdev.so

Installed Directories: /usr/include/umockdev-1.0

Short Descriptions

<code>umockdev-record</code>	records Linux devices and their ancestors from sysfs/udev or records ioctls for a device
<code>umockdev-run</code>	runs a program under an umockdev testbed
<code>umockdev-wrapper</code>	wraps a program around <code>libumockdev-preload.so.0</code> through LD_PRELOAD
<code>libumockdev.so</code>	provides API functions that allow mocking hardware devices for unit testing

utfcpp-4.0.5

Introduction to utfcpp

The utfcpp package contains a set of include files to provide UTF-8 with C++ in a Portable Way.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/nemtrif/utfcpp/archive/v4.0.5/utfcpp-4.0.5.tar.gz>
- Download MD5 sum: 8e0fe13266a7fa02f61340bf399986c3
- Download size: 36 KB
- Estimated disk space required: 496 KB
- Estimated build time: less than 0.1 SBU

utfcpp Dependencies

Required

[CMake-3.30.2](#)

Installation of utfcpp

Install utfcpp by running the following commands:

```
mkdir build &&
cd build &&
cmake -D CMAKE_INSTALL_PREFIX=/usr ..
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/include/utfcpp and /usr/share/utfcpp

Wayland-1.23.0

Introduction to Wayland

Wayland is a project to define a protocol for a compositor to talk to its clients as well as a library implementation of the protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/wayland/wayland/-/releases/1.23.0/downloads/wayland-1.23.0.tar.xz>
- Download MD5 sum: 23ad991e776ec8cf7e58b34cbd2efa75
- Download size: 236 KB
- Estimated disk space required: 6.8 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Wayland Dependencies

Required

[libxml2-2.13.3](#)

Optional

[Doxygen-1.12.0](#), [Graphviz-12.1.0](#) and [xmlto-0.0.29](#) (to build the API documentation) and [docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#) and [libxslt-1.1.42](#) (to build the manual pages)

Installation of Wayland

Install Wayland by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D documentation=false &&
ninja
```

To test the results, issue: `env -u XDG_RUNTIME_DIR ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D documentation=false`: This switch is used to disable building of the API documentation. Remove it if you have installed optional dependencies.

Contents

Installed Programs: wayland-scanner

Installed Libraries: libwayland-client.so, libwayland-cursor.so, libwayland-egl.so, and libwayland-server.so

Installed Directories: /usr/share/wayland

Short Descriptions

wayland-scanner	is a tool to generate proxy methods in wayland-client-protocol.h and wayland-server-protocol.h
libwayland-client.so	contains API functions for writing Wayland applications
libwayland-cursor.so	contains API functions for managing cursors in Wayland applications
libwayland-egl.so	contains API functions for handling OpenGL calls in Wayland applications

Wayland-Protocols-1.36

Introduction to Wayland-Protocols

The Wayland-Protocols package contains additional Wayland protocols that add functionality outside of protocols already in the Wayland core.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/wayland/wayland-protocols/-/releases/1.36/downloads/wayland-protocols-1.36.tar.xz>
- Download MD5 sum: d733380202a75ca837744e65b4dbadc5
- Download size: 96 KB
- Estimated disk space required: 8.9 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Wayland-protocols Dependencies

Required

[Wayland-1.23.0](#)

Installation of Wayland-protocols

Install Wayland-protocols by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/wayland-protocols

wv-1.2.9

Introduction to wv

The wv package contains tools for reading information from an MS Word document.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/wv/wv-1.2.9.tar.gz>
- Download MD5 sum: dbccf2e9f747e50c913b7e3d126b73f7
- Download size: 608 KB
- Estimated disk space required: 25 MB

- Estimated build time: 0.4 SBU

wv Dependencies

Required

[libgsf-1.14.52](#) and [libpng-1.6.43](#)

Optional

[libwmf](#)

Installation of wv

Install wv by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not have a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: wvSummary and several other wv* programs which are deprecated in favour of abiword: see <https://wvware.sourceforge.net/>

Installed Library: libwv-1.2.so

Installed Directory: /usr/share/wv

Short Descriptions

<code>wvSummary</code>	displays the summary information from an MS Word document
<code>libwv-1.2.so</code>	provides functions to access MS Word documents

Xapian-1.4.26

Introduction to xapian

Xapian is an open source search engine library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://oligarchy.co.uk/xapian/1.4.26/xapian-core-1.4.26.tar.xz>
- Download MD5 sum: 973fe094231c50a76c68f8f29fce4a27
- Download size: 3.1 MB
- Estimated disk space required: 146 MB (add 169 MB for tests)
- Estimated build time: 0.5 SBU (add 11 SBU for tests; both using parallelism=4)

Xapian Dependencies

Optional

[Valgrind-3.23.0](#) (for tests)

Installation of Xapian

Install Xapian by running the following commands:

```
./configure --prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/xapian-core-1.4.26 &&
make
```

To run the test suite, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `copydatabase`, `quest`, `simpleexpand`, `simpleindex`, `simplesearch`, `xapian-check`, `xapian-compact`, `xapian-config`, `xapian-delve`, `xapian-metadata`, `xapian-pos`, `xapian-progsrv`, `xapian-replicate`, `xapian-replicate-server` and `xapian-tcpsrv`

Installed Libraries: `libxapian.so`

Installed Directories: `/usr/include/xapian`, `/usr/lib/cmake/xapian`, `/usr/share/doc/xapian-core-1.4.26`, and `/usr/share/xapian-core`

Short Descriptions

<code>copydatabase</code>	performs a document-by-document copy of one or more Xapian databases
<code>quest</code>	is a command line tool to search through a database
<code>simpleexpand</code>	is a simple example program demonstrating query expansion
<code>simpleindex</code>	indexes each paragraph of a text file as a Xapian document
<code>simplesearch</code>	is a simple command line search utility
<code>xapian-check</code>	checks the consistency of a database or table
<code>xapian-compact</code>	compacts a database, or merges and compacts several databases
<code>xapian-config</code>	reports information about the installed version of xapian
<code>xapian-delve</code>	inspects the contents of a Xapian database
<code>xapian-metadata</code>	reads and writes user metadata
<code>xapian-pos</code>	inspects the contents of a flint table for development or debugging
<code>xapian-progsrv</code>	is a remote server for use with ProgClient
<code>xapian-replicate</code>	replicates a database from a master server to a local copy
<code>xapian-replicate-server</code>	services database replication requests from clients
<code>xapian-tcpsrv</code>	is the TCP daemon for use with Xapian's remote backend

Chapter 10. Graphics and Font Libraries

Depending on what your system will be used for, you may or may not require the graphics and font libraries. Most desktop machines will want them for use with graphical applications. Most servers on the other hand, will not require them.

AAlib-1.4rc5

Introduction to AAlib

AAlib is a library to render any graphic into ASCII Art.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/aa-project/aalib-1.4rc5.tar.gz>
- Download MD5 sum: 9801095c42bba12edebd1902bcf0a990
- Download size: 388 KB
- Estimated disk space required: 6.5 MB

- Estimated build time: 0.1 SBU

AAlib Dependencies

Optional

[Xorg Libraries](#), [Xorg Fonts](#) (runtime), [slang-2.3.3](#), and [GPM-1.20.7](#)

Installation of AAlib

Fix a minor problem with the included m4 file:

```
sed -i -e '/AM_PATH_AALIB,/s/AM_PATH_AALIB/[&]/' aalib.m4
```

Change the default X11 font from [Xorg Legacy Fonts](#) to [Xorg Fonts](#):

```
sed -e 's/8x13bold--*-luxi mono-bold-r-normal--13-120-*-*-*-*-/` \ 
-i src/aax.c
```

Fix an overuse of some ncurses internal data structures to allow building this package with ncurses-6.5 or later:

```
sed 's/stdscr->_max\(([xy])\) + 1/getmax\1(stdscr)/' \
-i src/aacurses.c
```

To allow building this package with GCC-14 or later, add some missing `#include` directives and fix a bad `return` statement to make the code C99-compatible. Then regenerate the `configure` script to ensure the C code for probing system features is C99-compatible as well:

```
sed -i 'li#include <stdlib.h>' \
src/aa{fire,info,lib,linuxkbd,savefont,test,regist}.c &&
sed -i 'li#include <string.h>' \
src/aa{kbddreg,moureg,test,regist}.c &&
sed -i '/X11_KBDDRIVER/a#include <X11/Xutil.h>' \
src/aaxkbd.c &&
sed -i '/rawmode_init/,/^}/s/return;/return 0;/' \
src/aalinuxkbd.c &&
autoconf
```

Install AAlib by running the following commands:

```
./configure --prefix=/usr \
--infodir=/usr/share/info \
--mandir=/usr/share/man \
--with-ncurses=/usr \
--disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: aafire, aainfo, aalib-config, aasavefont, and aatest

Installed Library: libaa.so

Installed Directories: None

Short Descriptions

aafire	is a demo of AAlib , rendering an animated fire in ASCII Art
aainfo	provides information for your current settings related to AAlib
aalib-	provides configuration info for AAlib
config	
aasavefont	saves a font to a file
aatest	shows the abilities of AAlib in a little test

babl-0.1.108

Introduction to Babl

The Babl package is a dynamic, any to any, pixel format translation library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gimp.org/pub/babl/0.1/babl-0.1.108.tar.xz>
- Download MD5 sum: 749169721b551882332a64ac17735de6
- Download size: 312 KB
- Estimated disk space required: 13 MB (with tests)
- Estimated build time: 0.1 SBU (Using parallelism=4; with tests)

Babl Dependencies

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [librsvg-2.58.3](#)

Optional

[Little CMS-2.16](#) and [w3m](#)

Installation of Babl

Install Babl by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install &&

install -v -m755 -d /usr/share/gtk-doc/html/babl/graphics &&
install -v -m644 docs/*.{css,html} /usr/share/gtk-doc/html/babl &&
install -v -m644 docs/graphics/*.{html,svg} /usr/share/gtk-doc/html/babl/graphics
```

Command Explanations

`install -v -m755 -d /usr/share/gtk-doc/html/babl/graphics`: This and the subsequent commands install the library html documentation under `/usr/share/gtk-doc/html` where other gtk packages put the programmer-oriented documentation.

Contents

Installed Programs: None

Installed Libraries: libbabl-0.1.so and libraries in /usr/lib/babl-0.1

Installed Directories: /usr/{include,lib}/babl-0.1 and /usr/share/gtk-doc/html/babl

Short Descriptions

libbabl-0.1.so contains functions to access BablFishes to convert between formats

Exiv2-0.28.3

Introduction to Exiv2

Exiv2 contains a C++ library and a command line utility for managing image and video metadata.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Exiv2/exiv2/archive/v0.28.3/exiv2-0.28.3.tar.gz>
- Download MD5 sum: 9944863400a43e79583b790f710b86c2
- Download size: 44 MB
- Estimated disk space required: 133 MB (with tests)
- Estimated build time: 0.3 SBU (Using parallelism=4, with tests)

Exiv2 dependencies

Required

[CMake-3.30.2](#)

Recommended

[Brotli-1.1.0](#), [cURL-8.9.1](#), and [injh-58](#)

Optional

[libssh](#)

Optional for documentation

[Doxygen-1.12.0](#), [Graphviz-12.1.0](#), and [libxslt-1.1.42](#)

Installation of Exiv2

Install Exiv2 by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D EXIV2_ENABLE_VIDEO=yes \
      -D EXIV2_ENABLE_WEBREADY=yes \
      -D EXIV2_ENABLE_CURL=yes \
      -D EXIV2_BUILD_SAMPLES=no \
      -D CMAKE_SKIP_INSTALL_RPATH=ON \
      -G Ninja ... &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D EXIV2_ENABLE_VIDEO=yes`: This switch enables managing video metadata.

`-D EXIV2_ENABLE_WEBREADY=yes`: This switch enables managing web image metadata.

`-D EXIV2_BUILD_SAMPLES=no`: This switch is necessary to suppress building and installing sample programs. If the sample programs are built, 34 additional programs are installed in /usr/bin.

`-D EXIV2_ENABLE_CURL=yes`: This switch is necessary to enable network/http capabilities.

`-D EXIV2_ENABLE_INI=no`: Use this switch if you have not installed [inih-58](#).

`-D EXIV2_ENABLE_BROTLI=no`: Use this switch if you have not installed [Brotli-1.1.0](#).

Contents

Installed Program: exiv2

Installed Library: libexiv2.so

Installed Directories: /usr/include/exiv2 and /usr/lib/cmake/exiv2

Short Descriptions

`exiv2` is a utility used to dump Exif data

FreeType-2.13.3

Introduction to FreeType2

The FreeType2 package contains a library which allows applications to properly render TrueType fonts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/freetype/freetype-2.13.3.tar.xz>
- Download MD5 sum: f3b4432c4212064c00500e1ad63fb64
- Download size: 2.5 MB
- Estimated disk space required: 33 MB (with additional documentation)
- Estimated build time: 0.2 SBU (with additional documentation)

Additional Downloads

Additional Documentation

- Download (HTTP): <https://downloads.sourceforge.net/freetype/freetype-doc-2.13.3.tar.xz>
- Download MD5 sum: 6affe0d431939398cc3c7cd58d824f8
- Download size: 2.1 MB

FreeType2 Dependencies

Recommended

[harfBuzz-9.0.0](#) (circular: build freetype, then harfbuzz, then reinstall freetype), [libpng-1.6.43](#), and [Which-2.21](#)

Optional

[Brotli-1.1.0](#) and [librsvg-2.58.3](#)

Optional (for documentation)

[docwriter](#)

Installation of FreeType2

If you downloaded the additional documentation, unpack it into the source tree using the following command:

```
tar -xf ../freetype-doc-2.13.3.tar.xz --strip-components=2 -C docs
```

Install FreeType2 by running the following commands:

```

sed -ri "s:.*(\AUX_MODULES.*valid):\\1:" modules.cfg &&
sed -r "s:.*(\#.*SUBPIXEL_RENDERING) .*:\\1:" \
-i include/freetype/config/ftoption.h &&
./configure --prefix=/usr --enable-freetype-config --disable-static &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you downloaded the optional documentation, install it as the `root` user:

```

cp -v -R docs -T /usr/share/doc/freetype-2.13.3 &&
rm -v /usr/share/doc/freetype-2.13.3/freetype-config.1

```

Command Explanations

`sed -ri ...`: First command enables GX/AAT and OpenType table validation and second command enables Subpixel Rendering. Note that Subpixel Rendering may have patent issues. Be sure to read the 'Other patent issues' part of <https://freetype.org/patents.html> before enabling this option.

`--enable-freetype-config`: This switch ensure that the man page for freetype-config is installed.

`--without-harfbuzz`: If harfbuzz is installed prior to freetype without freetype support, use this switch to avoid a build failure.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: freetype-config

Installed Library: libfreetype.so

Installed Directories: /usr/include/freetype2 and /usr/share/doc/freetype-2.13.3

Short Descriptions

`freetype-config` is used to get FreeType compilation and linking information

`libfreetype.so` contains functions for rendering various font types, such as TrueType and Type1

Fontconfig-2.15.0

Introduction to Fontconfig

The Fontconfig package contains a library and support programs used for configuring and customizing font access.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/fontconfig/release/fontconfig-2.15.0.tar.xz>
- Download MD5 sum: 5bb3a2829aecb22ae553c39099bd0d6a
- Download size: 1.4 MB
- Estimated disk space required: 17 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

Fontconfig Dependencies

Required

[FreeType-2.13.3](#)

Optional

[bubblewrap-0.9.0](#) (used by some tests), [cURL-8.9.1](#) and [UnZip-6.0](#) (both used by some tests for downloading and extracting test files), [JSON-C-0.17](#), [DocBook-utils-0.6.14](#) and [libxml2-2.13.3](#), [texlive-20240312](#) (or [install-tl-unx](#))

Note

An Internet connection is needed for some tests of this package.

Note

If you have DocBook Utils installed and you remove the `--disable-docs` parameter from the `configure` command below, you must have [SGMLPm-1.1](#) and [texlive-20240312](#) installed also, or the Fontconfig build will fail.

Installation of Fontconfig

Install Fontconfig by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --localstatedir=/var \
            --disable-docs \
            --docdir=/usr/share/doc/fontconfig-2.15.0 &&
make
```

To test the results, issue: `make check`. One test is known to fail if the kernel does not support user namespaces. Some tests will download some font files via Internet.

Now, as the `root` user:

```
make install
```

If you did not remove the `--disable-docs` parameter from the `configure` command, you can install the pre-generated documentation by using the following commands as the `root` user:

```
install -v -dm755 \
        /usr/share/{man/man{1,3,5},doc/fontconfig-2.15.0/fontconfig-devel} &&
install -v -m644 fc-*/*.1          /usr/share/man/man1 &&
install -v -m644 doc/*.3          /usr/share/man/man3 &&
install -v -m644 doc/fonts-conf.5 /usr/share/man/man5 &&
install -v -m644 doc/fontconfig-devel/* \
        /usr/share/doc/fontconfig-2.15.0/fontconfig-devel &&
install -v -m644 doc/*.{pdf,sgml,txt,html} \
        /usr/share/doc/fontconfig-2.15.0
```

Command Explanations

`--disable-docs`: This switch avoids building the documentation (the release tarball includes pre-generated documentation).

Configuring Fontconfig

Config Files

`/etc/fonts/*`, `/etc/fonts/conf.d/*` and `/usr/share/fontconfig/conf.avail/*`

Configuration Information

The main configuration file for Fontconfig is `/etc/fonts/fonts.conf`. Generally you do not want to edit this file. It will also read `/etc/fonts/local.conf` and any files in `/etc/fonts/conf.d`. To put a new font directory in the configuration, create (or update) the `/etc/fonts/local.conf` file with your local information or add a new file in `/etc/fonts/conf.d`. The default location of fonts in Fontconfig is:

- `/usr/share/fonts`
- `~/.local/share/fonts`
- `~/.fonts` (*this is now deprecated, but for the moment it still works*)

Fontconfig also ships many example configuration files in the `/usr/share/fontconfig/conf.avail` directory. Symlinking specific files to `/etc/fonts/conf.d` will enable them. The default setup is generally good enough for most users. See `/etc/fonts/conf.d/README` for a description of the configuration files.

More information about configuring Fontconfig can be found in the user's manual in <file:///usr/share/doc/fontconfig-2.15.0/fontconfig-user.html>.

Contents

Installed Programs: `fc-cache`, `fc-cat`, `fc-conflict`, `fc-list`, `fc-match`, `fc-pattern`, `fc-query`, `fc-scan`, and `fc-validate`

Installed Library: `libfontconfig.so`

Installed Directories: `/etc/fonts`, `/usr/include/fontconfig`, `/usr/share/doc/fontconfig-2.15.0`, `/usr/share/fontconfig`, `/usr/share/xml/fontconfig`, and `/var/cache/fontconfig`

Short Descriptions

<code>fc-cache</code>	is used to create font information caches
<code>fc-cat</code>	is used to read font information caches
<code>fc-conflict</code>	shows the ruleset files' information on the system
<code>fc-list</code>	is used to create font lists
<code>fc-match</code>	is used to match available fonts, or find fonts that match a given pattern
<code>fc-pattern</code>	is used to parse pattern (empty pattern by default) and show the parsed result
<code>fc-query</code>	is used to query fonts files and print resulting patterns
<code>fc-scan</code>	is used to scan font files and directories, and print resulting patterns
<code>fc-validate</code>	is used to validate font files
<code>libfontconfig.so</code>	contains functions used by the <code>Fontconfig</code> programs and also by other programs to configure or customize font access

FriBidi-1.0.15

Introduction to FriBidi

The FriBidi package is an implementation of the [Unicode Bidirectional Algorithm \(BIDI\)](#). This is useful for supporting Arabic and Hebrew alphabets in other packages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/fribidi/fribidi/releases/download/v1.0.15/fribidi-1.0.15.tar.xz>
- Download MD5 sum: `ccf2b019162b4e5e6569875d0641bc5f`
- Download size: 1.1 MB
- Estimated disk space required: 21 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

FriBidi Dependencies

Optional

[c2man \(to build man pages\)](#)

Installation of FriBidi

Install FriBidi by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: fribidi

Installed Library: libfribidi.so

Installed Directory: /usr/include/fribidi

Short Descriptions

fribidi is a command-line interface to the libfribidi library and can be used to convert a logical string to visual output

libfribidi.so contains functions used to implement the [Unicode Bidirectional Algorithm](#)

gegl-0.4.48

Introduction to gegl

This package provides the GEneric Graphics Library, which is a graph based image processing format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gimp.org/pub/gegl/0.4/gegl-0.4.48.tar.xz>
- Download MD5 sum: a96ae32d46855ab4eefa1fce2dbefda1
- Download size: 5.5 MB
- Estimated disk space required: 80 MB (Add 4 MB for tests)
- Estimated build time: 0.7 SBU (add 0.1 SBU for tests, both with parallelism=4)

gegl Dependencies

Required

[babl-0.1.108](#) and [JSON-GLib-1.8.0](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [Graphviz-12.1.0](#) (build with pango and libpng), [Pygments-2.18.0](#), and [PyGObject-3.48.2](#)

Optional

[asciidoc-10.2.1](#), [Cairo-1.18.0](#), [FFmpeg-7.0.2](#) (currently broken), [gdk-pixbuf-2.42.12](#), [gexiv2-0.14.3](#), [GTK-Doc-1.34.0](#), [JasPer-4.2.4](#), [Little CMS-2.16](#), [libraw-0.21.2](#), [librsvg-2.58.3](#), [libspiro-20220722](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [Pango-1.54.0](#), [Poppler-24.08.0](#), [Ruby-3.3.4](#), [SDL2-2.30.6](#), [v4l-utils-1.28.1](#), [Vala-0.56.17](#), [LuaJIT](#), [lensfun](#), [LuaJIT](#), [libnsgif](#), [libumfpack](#), [maxflow](#), [MRG](#), [OpenCL](#), [OpenEXR](#), [poly2tri-c](#), [source-highlight](#), and [w3m](#)

Installation of gegl

If you are installing over a previous version of gegl, one of the modules will need to be removed. As the `root` user, run the following command to remove it:

```
rm -f /usr/lib/gegl-0.4/vector-fill.so
```

Install gegl by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D libav=disabled .. &&
ninja
```

To run the tests, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D libav=disabled`: This switch disables building support for ffmpeg because the latest version causes the build to fail.

`-D docs=true`: Use this switch to build and install the documentation (requires [GTK-Doc-1.34.0](#)).

Contents

Installed Programs: gegl and gegl-imgcmp

Installed Libraries: libgegl-0.4.so, libgegl-npd-0.4.so, and modules in /usr/lib/gegl-0.4

Installed Directories: /usr/lib/gegl-0.4 and /usr/include/gegl-0.4

Short Descriptions

<code>gegl</code>	is a commandline tool for interfacing with the gegl library
<code>gegl-imgcmp</code>	is a simple image difference detection tool for use in regression testing
<code>libgegl-0.4.so</code>	provides infrastructure to do demand based cached non destructive image editing on larger than RAM buffers
<code>libgegl-npd-0.4.so</code>	is the GEGL N-point image deformation library

giflib-5.2.2

Introduction to giflib

The giflib package contains libraries for reading and writing GIFs as well as programs for converting and working with GIF files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sourceforge.net/projects/giflib/files/giflib-5.2.2.tar.gz>
- Download MD5 sum: 913dd251492134e235ee3c9a91987a4d
- Download size: 440 KB
- Estimated disk space required: 4.0 MB (with documentation)
- Estimated build time: less than 0.1 SBU (with documentation)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/giflib-5.2.2-upstream_fixes-1.patch

giflib Dependencies

Required

[xmlto-0.0.29](#)

Installation of giflib

First, prevent the build process from installing XML files instead of man pages:

```
patch -Np1 -i ../giflib-5.2.2-upstream_fixes-1.patch
```

Next, remove an unnecessary dependency on [ImageMagick-7.1.1-36](#) by moving a file into an expected location:

```
cp pic/gifgrid.gif doc/giflib-logo.gif
```

Install giflib by running the following commands:

```
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr install &&  
  
rm -fv /usr/lib/libgif.a &&  
  
find doc \( -name Makefile\* -o -name \*.1 \  
-o -name \*.xml \) -exec rm -v {} \; &&  
  
install -v -dm755 /usr/share/doc/giflib-5.2.2 &&  
cp -v -R doc/* /usr/share/doc/giflib-5.2.2
```

Command Explanations

`rm -fv /usr/lib/libgif.a`: This command removes a static library which is not used by any BLFS package.

`find doc ... -exec rm -v {} \;`: This command removes `Makefiles`, man and xml files from the documentation directory that would otherwise be installed by the commands that follow.

Contents

Installed Programs: gif2rgb, gifbuild, gifclrmp, giffix, giftext, and giftool

Installed Library: libgif.so

Installed Directory: /usr/share/doc/giflib-5.2.2

Short Descriptions

<code>gif2rgb</code>	converts images saved as GIF to 24-bit RGB images
<code>gifbuild</code>	dumps GIF data in a textual format, or undumps it to a GIF
<code>gifclrmp</code>	modifies GIF image colormaps
<code>giffix</code>	clumsily attempts to fix truncated GIF images
<code>giftext</code>	prints (text only) general information about a GIF file
<code>giftool</code>	is a GIF transformation tool
<code>libgif.so</code>	contains API functions required by the <code>giflib</code> programs and any other programs needing library functionality to read, write and manipulate GIF images

Glad-2.0.6

Introduction to Glad

The Glad package contains a generator for loading Vulkan, OpenGL, EGL, GLES, and GLX contexts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Dav1dde/glad/archive/v2.0.6/glad-2.0.6.tar.gz>
- Download MD5 sum: 350c34126e47988f378285adf4cf2efc
- Download size: 620 KB
- Estimated disk space required: 14 MB
- Estimated build time: less than 0.1 SBU

Glad Dependencies

Optional (required to run the tests)

[pytest-8.3.2](#), [rustc-1.80.1](#), [Xorg Libraries](#), [glfw](#), and [WINE](#)

Installation of Glad

Install Glad by running the following commands:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package comes with a test suite, but it cannot be run without installing the external dependencies listed above.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user glad2
```

Contents

Installed Programs: glad

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/glad and /usr/lib/python3.12/site-packages/glad2-2.0.6.dist-info

Short Descriptions

glad generates loaders for Vulkan, OpenGL, EGL, GLES, and GLX contexts

GLM-1.0.1

Introduction to GLM

OpenGL Mathematics (GLM) is a header-only C++ mathematics library for graphics software based on the OpenGL Shading Language (GLSL) specifications. An extension system provides extended capabilities such as matrix transformations and quaternions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/g-truc/glm/archive/1.0.1/glm-1.0.1.tar.gz>
- Download MD5 sum: f824ac50e16310a95279032f82cbd341
- Download size: 4.4 MB
- Estimated disk space required: 44 MB
- Estimated build time: less than 0.1 SBU

Installation of GLM

Note

This package is unusual as it includes its functionality in header files. We just copy them into position.

As the `root` user:

```
cp -r glm /usr/include/ &&
cp -r doc /usr/share/doc/glm-1.0.1
```

Contents

Installed Program: None

Installed Library: None

Graphite2-1.3.14

Introduction to Graphite2

Graphite2 is a rendering engine for graphite fonts. These are TrueType fonts with additional tables containing smart rendering information and were originally developed to support complex non-Roman writing systems. They may contain rules for e.g. ligatures, glyph substitution, kerning, justification - this can make them useful even on text written in Roman writing systems such as English. Note that firefox by default provides an internal copy of the graphite engine and cannot use a system version (although it can now be patched to use it), but it too should benefit from the availability of graphite fonts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/silnrsi/graphite/releases/download/1.3.14/graphite2-1.3.14.tgz>
- Download MD5 sum: 1bccb985a7da01092bfb53bb5041e836
- Download size: 6.3 MB
- Estimated disk space required: 30 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Graphite2 Dependencies

Required

[CMake-3.30.2](#)

Optional

[FreeType-2.13.3](#), [silgraphite](#) to build the `comparerender` test and benchmarking tool, and if that is present, and [harfBuzz-9.0.0](#) to add more functionality to it (this is a circular dependency, you would need to first build graphite2 without harfbuzz).

To build the documentation: [asciidoc-10.2.1](#), [Doxygen-1.12.0](#), [texlive-20240312](#) (or [install-tl-unx](#)), and [dblatex](#) (for PDF docs)

To execute the test suite you will need [FontTools](#) (Python 3 module), otherwise, the "cmp" tests fail.

Optional (at runtime)

You will need at least one suitable [graphite font](#) for the package to be useful.

Installation of Graphite2

Some tests fail if [FontTools](#) (Python 3 module) is not installed. These tests can be removed with:

```
sed -i '/cmptest/d' tests/CMakeLists.txt
```

Install Graphite2 by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr .. &&
make
```

If you wish to build the documentation, issue:

```
make docs
```

To test the results, issue: `make test`. One test named `nametabletest` is known to fail.

Now, as the `root` user:

```
make install
```

If you built the documentation, install, as the `root` user:

```
install -v -d -m755 /usr/share/doc/graphite2-1.3.14 &&
cp      -v -f    doc/{GTF,manual}.html \
           /usr/share/doc/graphite2-1.3.14 &&
cp      -v -f    doc/{GTF,manual}.pdf \
           /usr/share/doc/graphite2-1.3.14
```

Command Explanations

`-D CMAKE_VERBOSE_MAKEFILE=ON`: This switch turns on build verbose mode.

Contents

Installed Programs: gr2fonttest, and optionally comparerender

Installed Libraries: libgraphite2.so

Installed Directories: /usr/{include,share}/graphite2 and optionally /usr/share/doc/graphite2-1.3.14

Short Descriptions

<code>comparerender</code>	is a test and benchmarking tool
<code>gr2fonttest</code>	is a diagnostic console tool for graphite fonts
<code>libgraphite2.so</code>	is a rendering engine for graphite fonts

harfBuzz-9.0.0

Introduction to Harfbuzz

The HarfBuzz package contains an OpenType text shaping engine.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/harfbuzz/harfbuzz/releases/download/9.0.0/harfbuzz-9.0.0.tar.xz>
- Download MD5 sum: 0035c129cb1646ab1cff65e5ef7153db
- Download size: 17 MB
- Estimated disk space required: 138 MB (With tests)
- Estimated build time: 0.7 SBU (With tests; both using parallelism=4)

HarfBuzz Dependencies

Recommended

[GLib-2.80.4](#) (required for Pango; GObject Introspection required for building GNOME), [Graphite2-1.3.14](#) (required for building [texlive-20240312](#) or [LibreOffice-24.8.0.3](#) with system harfbuzz), [ICU-75.1](#), and [FreeType-2.13.3](#) (after harfbuzz is installed, reinstall freetype)

Optional

[Cairo-1.18.0](#) (circular: build cairo and all its recommended dependencies, including harfbuzz, first, then rebuild harfbuzz if the cairo backend is needed), [git-2.46.0](#), [GTK-Doc-1.34.0](#), [FontTools](#) (Python 3 module, for the test suite), [raged](#), and [wasm-micro-runtime](#)

Warning

Recommended dependencies are not strictly required to build the package. However, you might not get expected results at runtime if you don't install them. Please do not report bugs with this package if you **have not** installed the recommended dependencies.

Installation of HarfBuzz

Install HarfBuzz by running the following commands:

```
mkdir build &&
cd build &&

meson setup ..           \
    --prefix=/usr          \
    --buildtype=release    \
    -D graphite2=enabled &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D graphite2=enabled`: This switch enables Graphite2 support, which is required for building [texlive-20240312](#) or [LibreOffice-24.8.0.3](#) with system harfbuzz.

`-D docs=disabled`: If [GTK-Doc-1.34.0](#) is installed, the documentation is built and installed. This switch prevents that.

Contents

Installed Programs: `hb-info`, `hb-ot-shape-closure`, `hb-shape`, `hb-subset`, and `hb-view` (only if Cairo is installed)

Installed Libraries: `libharfbuzz.so`, `libharfbuzz-cairo.so` (only if Cairo is installed), `libharfbuzz-gobject.so`, `libharfbuzz-icu.so`, and `libharfbuzz-subset.so`

Installed Directories: `/usr/include/harfbuzz`, `/usr/lib/cmake/harfbuzz`, and `/usr/share/gtk-doc/html/harfbuzz` (optional)

Short Descriptions

<code>hb-info</code>	is used for gathering information about fonts installed on the system
<code>hb-ot-shape-closure</code>	gives the set of characters contained in a string, represented as single characters and/or single character names. Example: <code>hb-ot-shape-closure /usr/share/fonts/dejavu/DejaVuSans.ttf "Hello World."</code>
<code>hb-shape</code>	is used for the conversion of text strings into positioned glyphs
<code>hb-subset</code>	is used to create subsets of fonts, and display text using them
<code>hb-view</code>	displays a graphical view of a string shape using a particular font as a set of glyphs. The output format is automatically defined by the file extension, the supported ones being ansi/png/svg/pdf/ps/eps. For example: <code>hb-view --output-file=hello.png /usr/share/fonts/dejavu/DejaVuSans.ttf "Hello World."</code>
<code>libharfbuzz.so</code>	is the HarfBuzz text shaping library
<code>libharfbuzz-cairo.so</code>	provides Cairo integration for the HarfBuzz text shaping library
<code>libharfbuzz-gobject.so</code>	provides GObject integration for the HarfBuzz text shaping library
<code>libharfbuzz-icu.so</code>	provides ICU integration for the HarfBuzz text shaping library
<code>libharfbuzz-subset.so</code>	provides API functions for performing subsetting operations on font files

JasPer-4.2.4

Introduction to JasPer

The JasPer Project is an open-source initiative to provide a free software-based reference implementation of the JPEG-2000 codec.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/jasper-software/jasper/archive/version-4.2.4/jasper-4.2.4.tar.gz>
- Download MD5 sum: aa4df693b90223fe6848b34cf1208624
- Download size: 1.9 MB
- Estimated disk space required: 9.4 MB (with tests)
- Estimated build time: 0.3 SBU (with tests)

JasPer Dependencies

Required

[CMake-3.30.2](#)

Recommended

[libjpeg-turbo-3.0.1](#)

Optional

[Freeglut-3.6.0](#) (required for jiv), [Doxygen-1.12.0](#) (needed for generating html documentation), and [texlive-20240312](#) (needed to regenerate the pdf documentation)

Installation of JasPer

Note

The tarball *jasper-4.2.4.tar.gz* will extract to *jasper-version-4.2.4*.

Install JasPer by running the following commands:

```
mkdir BUILD &&
cd BUILD &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_BUILD_TYPE=Release           \
-D CMAKE_SKIP_INSTALL_RPATH=ON        \
-D JAS_ENABLE_DOC=NO                 \
-D ALLOW_IN_SOURCE_BUILD=YES         \
-D CMAKE_INSTALL_DOCDIR=/usr/share/doc/jasper-4.2.4 \
.. &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D JAS_ENABLE_DOC=NO`: This option disables rebuilding the pdf documentation if [texlive-20240312](#) is installed.

`-D ALLOW_IN_SOURCE_BUILD=YES`: This switch allows building from within the source tree. In our case, this is needed to allow us to build inside of the BUILD directory instead of needing to create another directory outside of the source tree.

Contents

Installed Programs: imgcmp, imginfo, jasper, and jiv

Installed Library: libjasper.so

Installed Directories: /usr/include/jasper and /usr/share/doc/jasper-4.2.4

Short Descriptions

<code>imgcmp</code>	compares two images of the same geometry
<code>imginfo</code>	displays information about an image
<code>jasper</code>	converts images between formats (BMP, JPS, JPC, JPG, PGX, PNM, MIF, and RAS)
<code>jiv</code>	displays images
<code>libjasper.so</code>	is a library used by programs for reading and writing JPEG2000 format files

Little CMS-2.16

Introduction to Little CMS2

The Little Color Management System is a small-footprint color management engine, with special focus on accuracy and performance. It uses the International Color Consortium standard (ICC), which is the modern standard for color management.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mm2/Little-CMS/releases/download/lcms2.16/lcms2-2.16.tar.gz>
- Download MD5 sum: f219d87c247957c97020a3859d8d6fa8
- Download size: 7.3 MB
- Estimated disk space required: 22 MB (with the tests)
- Estimated build time: 0.2 SBU (with the tests)

Little CMS2 Dependencies

Optional

[libjpeg-turbo-3.0.1](#) and [libtiff-4.6.0](#)

Installation of Little CMS2

Install Little CMS2 by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: jpgicc, linkicc, psicc, tificc, and transicc

Installed Library: liblcms2.so

Installed Directories: None

Short Descriptions

<code>jpgicc</code>	is the Little CMS ICC profile applier for JPEG
<code>linkicc</code>	is the Little CMS ICC device link generator
<code>psicc</code>	is the Little CMS ICC PostScript generator
<code>tificc</code>	is the Little CMS ICC tiff generator
<code>transicc</code>	is the Little CMS ColorSpace conversion calculator

`liblcms2.so` contains functions that implement the lcms2 API

libavif-1.1.1

Introduction to libavif

The libavif package contains a library used for encoding and decoding AVIF files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/AOMediaCodec/libavif/archive/v1.1.1/libavif-1.1.1.tar.gz>
- Download MD5 sum: 633c2113d52aecab5f0073da3456e2ae
- Download size: 13 MB
- Estimated disk space required: 21 MB
- Estimated build time: less than 0.1 SBU

libavif Dependencies

Required

[libaom-3.9.1](#)

Recommended

[gdk-pixbuf-2.42.12](#)

Optional

[gtest](#), [libdav1d](#), [libyuv](#), [rav1e](#), and [svt-av1](#)

Note

An Internet connection is needed for some tests of this package.

Installation of libavif

Install libavif by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D AVIF_CODEC_AOM=SYSTEM \
-D AVIF_BUILD_GDK_PIXBUF=ON \
-D AVIF_LIBYUV=OFF \
-G Ninja .. &&
ninja
```

To test the package (note that it will cause the building system to download a copy of [gtest](#) and build the test suite with the copy), issue:

```
cmake .. -D AVIF_GTEST=LOCAL -D AVIF_BUILD_TESTS=ON &&
ninja && ninja test
```

Now, as the `root` user:

```
ninja install
```

The AV1 format needs to be added to the loaders cache. As the `root` user:

```
gdk-pixbuf-query-loaders --update-cache
```

Command Explanations

-D AVIF_CODEC_AOM=SYSTEM: This switch enables using the AOM codec. This package is useless without at least one codec built in.

-D AVIF_BUILD_GDK_PIXBUF=ON: This switch builds the AVIF loader for applications which use gdk-pixbuf. Remove it if you have not installed [gdk-pixbuf-2.42.12](#).

-D AVIF_LIBYUV=OFF: Use this switch if you have not installed [libyuv](#).

-D AVIF_CODEC_DAV1D=SYSTEM: Use this switch if you have installed [libdav1d](#) and wish to use it as a codec.

-D AVIF_CODEC_RAV1E=SYSTEM: Use this switch if you have installed [rav1e](#) and wish to use it as a codec.

-D AVIF_CODEC_SVT=SYSTEM: Use this switch if you have installed [svt-av1](#) and wish to use it as a codec.

Contents

Installed Programs: None

Installed Libraries: libavif.so and libpixbufloader-avif.so (in /usr/lib/gdk-pixbuf-2.0/2.10.0/loaders)

Installed Directories: /usr/include/avif and /usr/lib/cmake/libavif

Short Descriptions

libavif.so contains functions that provide a portable C implementation of the AV1 Image Format
libpixbufloader-avif.so allows applications which use gdk-pixbuf to read AVIF images

libexif-0.6.24

Introduction to libexif

The libexif package contains a library for parsing, editing, and saving EXIF data. Most digital cameras produce EXIF files, which are JPEG files with extra tags that contain information about the image. All EXIF tags described in EXIF standard 2.1 are supported.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libexif/libexif/releases/download/v0.6.24/libexif-0.6.24.tar.bz2>
- Download MD5 sum: 00ea6e7ff62d3fd41ea9b2139746703c
- Download size: 1.9 MB
- Estimated disk space required: 17 MB
- Estimated build time: 0.1 SBU

libexif Dependencies

Optional (to Build Documentation)

[Doxygen-1.12.0](#) and [Graphviz-12.1.0](#)

Installation of libexif

Install libexif by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --with-doc-dir=/usr/share/doc/libexif-0.6.24 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Documentation was built and installed if you have the dependencies shown above installed. If you don't have the dependencies installed, there is a compressed tarball in the source tree `doc` directory that can be unpacked into `/usr/share/doc/libexif-0.6.24`.

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libexif.so

Installed Directories: /usr/include/libexif and /usr/share/doc/libexif-0.6.24

Short Descriptions

`libexif.so` contains functions used for parsing, editing, and saving EXIF data

libgxpath-0.3.2

Introduction to libgxpath

The libgxpath package provides an interface to manipulate XPS documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgxpath/0.3/libgxpath-0.3.2.tar.xz>
- Download MD5 sum: 0527ac7c8c405445e96a5baa6019a0c3
- Download size: 80 KB
- Estimated disk space required: 5.4 MB
- Estimated build time: 0.1 SBU

Libgxpath Dependencies

Required

[GTK+-3.24.43](#), [Little CMS-2.16](#), [libarchive-3.7.4](#), [libjpeg-turbo-3.0.1](#), [libtiff-4.6.0](#), and [libxslt-1.1.42](#)

Optional

[git-2.46.0](#) and [GTK-Doc-1.34.0](#)

Installation of Libgxpath

Install Libgxpath by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: xpstojpeg, xpstopdf, xpstopng, xpstops, and xpstosvg

Installed Library: libgxpath.so

Installed Directories: /usr/include/libgxpath

Short Descriptions

xpstojpeg	converts XPS documents to a JPEG image
xpstopdf	converts XPS documents to PDF format
xpstopng	converts XPS documents to a PNG image
xpstops	converts XPS documents to PostScript
xpstosvg	converts XPS documents to SVG images
libgxpath.so	contains API functions for manipulating XPS documents

libjpeg-turbo-3.0.1

Introduction to libjpeg-turbo

libjpeg-turbo is a fork of the original IJG libjpeg which uses SIMD to accelerate baseline JPEG compression and decompression. libjpeg is a library that implements JPEG image encoding, decoding and transcoding.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/libjpeg-turbo/libjpeg-turbo-3.0.1.tar.gz>
- Download MD5 sum: 1fdc6494521a8724f5f7cf39b0f6aff3
- Download size: 2.7 MB
- Estimated disk space required: 55 MB (with tests)
- Estimated build time: 0.5 SBU (with tests; both using parallelism=4)

libjpeg-turbo Dependencies

Required

[CMake-3.30.2](#)

Recommended

[NASM-2.16.03](#) or [yasm-1.3.0](#) (for building the package with optimized assembly routine)

Installation of libjpeg-turbo

Install libjpeg-turbo by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_BUILD_TYPE=RELEASE          \
-D ENABLE_STATIC=FALSE              \
-D CMAKE_INSTALL_DEFAULT_LIBDIR=lib \
-D CMAKE_SKIP_INSTALL_RPATH=ON      \
-D CMAKE_INSTALL_DOCDIR=/usr/share/doc/libjpeg-turbo-3.0.1 \
.. &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D WITH_JPEG8=ON`: This switch enables compatibility with libjpeg version 8.

Contents

Installed Programs: `cjpeg`, `djpeg`, `jpegtran`, `rdjpgcom`, `tjbench`, and `wrjpgcom`

Installed Libraries: `libjpeg.so` and `libturbojpeg.so`

Installed Directories: `/usr/share/doc/libjpeg-turbo-3.0.1`

Short Descriptions

<code>cjpeg</code>	compresses image files to produce a JPEG/JFIF file on the standard output. Currently supported input file formats are: PPM (PBMPLUS color format), PGM (PBMPLUS gray-scale format), BMP, and Targa
<code>djpeg</code>	decompresses image files from JPEG/JFIF format to either PPM (PBMPLUS color format), PGM (PBMPLUS gray-scale format), BMP, or Targa format
<code>jpegtran</code>	is used for lossless transformation of JPEG files
<code>rdjpgcom</code>	displays text comments from within a JPEG file
<code>tjbench</code>	is used to benchmark the performance of libjpeg-turbo
<code>wrjpgcom</code>	inserts text comments into a JPEG file
<code>libjpeg.so</code>	contains functions used for reading and writing JPEG images

libjxl-0.10.3

Introduction to libjxl

The libjxl package contains the reference implementation of the JPEG XL image format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libjxl/libjxl/archive/v0.10.3/libjxl-0.10.3.tar.gz>
- Download MD5 sum: 0fd3db8956a41d13b5e8eac4fe61d8d3
- Download size: 1.8 MB
- Estimated disk space required: 42 MB
- Estimated build time: 0.6 SBU (with parallelism=4)

libjxl Dependencies

Required

[Brotli-1.1.0](#), [CMake-3.30.2](#), [glib-5.2.2](#), [highway-1.2.0](#), [Little CMS-2.16](#), [libjpeg-turbo-3.0.1](#), and [libpng-1.6.43](#)

Recommended

[gdk-pixbuf-2.42.12](#) (for the plugin)

Optional

[Doxygen-1.12.0](#) and [Graphviz-12.1.0](#) (for documentation), [Java-22.0.2](#) (for the JAR), [libavif-1.1.1](#), [libwebp-1.4.0](#), [gtest](#), [OpenEXR](#), [sjpeg](#), and [skcms](#)

Installation of libjxl

Install libjxl by running the following commands:

```

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr           \
      -D CMAKE_BUILD_TYPE=Release            \
      -D BUILD_TESTING=OFF                  \
      -D BUILD_SHARED_LIBS=ON                \
      -D JPEGXL_ENABLE_SKCMS=OFF             \
      -D JPEGXL_ENABLE_SJPEG=OFF              \
      -D JPEGXL_ENABLE_PLUGINS=ON             \
      -D JPEGXL_INSTALL_JARDIR=/usr/share/java \
      -G Ninja ..                           \
      &&
ninja

```

This package does come with a test suite, but it requires [gtest](#), which is not in BLFS.

Now, as the `root` user:

```

ninja install &&
gdk-pixbuf-query-loaders --update-cache

```

Command Explanations

`gdk-pixbuf-query-loaders --update-cache`: This command regenerates the GDK Pixbuf loader cache so that it is aware of the JPEG-XL loader.

`-D BUILD_TESTING=OFF`: This parameter disables the tests because they require [gtest](#), which is not in BLFS. If you wish to run the tests and have [gtest](#) installed, remove this parameter.

`-D BUILD_SHARED_LIBS=ON`: This parameter enables building shared versions of the libraries instead of static ones.

`-D JPEGXL_ENABLE_SKCMS=OFF`: This parameter disables building support for Skia's skcms since it is not necessary for BLFS purposes and requires [skcms](#).

`-D JPEGXL_ENABLE_SJPEG=OFF`: This parameter disables support for SimpleJPEG since it is not necessary for BLFS purposes and requires [sjpeg](#).

`-D JPEGXL_ENABLE_PLUGINS=ON`: This parameter enables Plugin support, which creates plugins for [gdk-pixbuf-2.42.12](#) if the package is installed.

`-D JPEGXL_INSTALL_JARDIR=/usr/share/java`: This parameter places the JAR file for JPEG XL support in the correct directory if [Java-22.0.2](#) is installed.

Contents

Installed Programs: `benchmark_xl`, `cjxl`, `djxl`, and `jxlinfo`

Installed Libraries: `libjxl.so`, `libjxl_cms.so`, `libjxl_extras_codec.so`, `libjxl_jni.so`, `libjxl_threads.so`, and `libpixbufloader-jxl.so` (in /usr/lib/gdk-pixbuf-2.0/2.10.0/loaders)

Installed Directories: /usr/include/jxl

Short Descriptions

<code>benchmark_xl</code>	runs performance benchmarks against libjxl
<code>cjxl</code>	compresses images into JPEG XL format
<code>djxl</code>	decompresses images from JPEG XL format into other formats
<code>jxlinfo</code>	displays information about JPEG XL images
<code>libjxl.so</code>	contains the reference implementation of the JPEG XL standard
<code>libjxl_cms.so</code>	contains support for Little CMS in libjxl
<code>libjxl_extras_codec.so</code>	contains additional codec support for JPEG XL images
<code>libjxl_jni.so</code>	contains a Java interface to support JPEG XL on systems where Java-22.0.2 was installed at build time
<code>libjxl_threads.so</code>	contains threading functions for JPEG XL
<code>libpixbufloader-jxl.so</code>	contains a loader for gdk-pixbuf to allow it to load JPEG XL images

Introduction to libmng

The libmng libraries are used by programs wanting to read and write Multiple-image Network Graphics (MNG) files which are the animation equivalents to PNG files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/libmng/libmng-2.0.3.tar.xz>
- Download MD5 sum: e9e899adb1b681b17f14d91e261878c5
- Download size: 932 KB
- Estimated disk space required: 15 MB
- Estimated build time: 0.1 SBU

libmng Dependencies

Required

[libjpeg-turbo-3.0.1](#) and [Little CMS-2.16](#)

Installation of libmng

Install libmng by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -v -m755 -d      /usr/share/doc/libmng-2.0.3 &&
install -v -m644 doc/*.txt /usr/share/doc/libmng-2.0.3
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libmng.so

Installed Directory: /usr/share/doc/libmng-2.0.3

Short Descriptions

libmng.so provides functions for programs wishing to read and write MNG files which are animation files without the patent problems associated with certain other formats

libmypaint-1.6.1

Introduction to libmypaint

The libmypaint package, a.k.a. "brushlib", is a library for making brushstrokes which is used by MyPaint and other projects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mypaint/libmypaint/releases/download/v1.6.1/libmypaint-1.6.1.tar.xz>
- Download MD5 sum: 7f1dab2d30ce8a3f494354c7c77a2977

- Download size: 508 KB
- Estimated disk space required: 11 MB (add 1 MB for tests)
- Estimated build time: 0.1 SBU (add 0.3 SBU for tests)

libmypaint Dependencies

Required

[JSON-C-0.17](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[Doxygen-1.12.0](#) (to create XML docs), [gegl \(0.3 versions only\)](#), and [gperf-tools](#)

Installation of libmypaint

Install libmypaint by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None.

Installed Libraries: libmypaint.so (and optionally libmypaint-gegl.so which is not used by any packages in this book).

Installed Directory: /usr/include/libmypaint

Short Descriptions

libmypaint.so contains functions for making brushstrokes

libpng-1.6.43

Introduction to libpng

The libpng package contains libraries used by other programs for reading and writing PNG files. The PNG format was designed as a replacement for GIF and, to a lesser extent, TIFF, with many improvements and extensions and lack of patent problems.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/libpng/libpng-1.6.43.tar.xz>
- Download MD5 sum: 22b8362d16c3724eba9c1fb8d187320a
- Download size: 1.0 MB
- Estimated disk space required: 15 MB (with tests)
- Estimated build time: 0.1 SBU (add 0.5 SBU for tests)

Additional Downloads

- Recommended patch to include animated png functionality in libpng (required to use the system libpng in Firefox, Seamonkey, and Thunderbird): <https://downloads.sourceforge.net/sourceforge/libpng-apng/libpng-1.6.43-apng.patch.gz>

- Patch md5sum: fc21268c9c583c5d2e7ead8a9ec17240

Installation of libpng

If you want to patch libpng to support apng files, apply it here:

```
gzip -cd ./libpng-1.6.43-apng.patch.gz | patch -p1
```

Install libpng by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
mkdir -v /usr/share/doc/libpng-1.6.43 &&
cp -v README libpng-manual.txt /usr/share/doc/libpng-1.6.43
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: libpng-config (symlink), libpng16-config, pngfix and png-fix-itxt

Installed Libraries: libpng.so

Installed Directories: /usr/include/libpng16 and /usr/share/doc/libpng-1.6.43

Short Descriptions

<code>pngfix</code>	tests, optimizes and optionally fixes the zlib header in PNG files. Optionally, when fixing, strips ancillary chunks from the file
<code>png-fix-itxt</code>	fixes PNG files that have an incorrect length field in the iTxt chunks
<code>libpng-config</code>	is a shell script that provides configuration information for applications wanting to use libpng
<code>libpng.so</code>	contain routines used to create and manipulate PNG format graphics files

libraw-0.21.2

Introduction to libraw

Libraw is a library for reading RAW files obtained from digital cameras (CRW/CR2, NEF, RAF, DNG, and others).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.libraw.org/data/LibRaw-0.21.2.tar.gz>
- Download MD5 sum: 0533724bad17c0fde22e642e4594e45d
- Download size: 1.6 MB
- Estimated disk space required: 30 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

libraw Dependencies

Recommended

[libjpeg-turbo-3.0.1](#), [JasPer-4.2.4](#), and [Little CMS-2.16](#)

Installation of libraw

Install libraw by running the following commands:

```
autoreconf -fiv      &&
./configure --prefix=/usr \
    --enable-jpeg \
    --enable-jasper \
    --enable-lcms \
    --disable-static \
    --docdir=/usr/share/doc/libraw-0.21.2 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-jpeg`: This switch enables support for JPEG images. Remove it if you don't have [libjpeg-turbo-3.0.1](#) installed.

`--enable-jasper`: This switch enables support for JPEG2000 images. Remove it if you don't have [JasPer-4.2.4](#) installed.

`--enable-lcms`: This switch enables support for color management. Remove it if you don't have [Little CMS-2.16](#) installed.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Note

All of the installed programs are examples of using libraw.

Installed Programs: `4channels`, `ddraw_emu`, `ddraw_half`, `half_mt`, `mem_image`, `multirender_test`, `postprocessing_benchmark`, `raw-identify`, `simple_ddraw`, and `unprocessed_raw`

Installed Library: `libraw.so` and `libraw_r.so`

Installed Directories: `/usr/include/libraw` and `/usr/share/doc/libraw-0.21.2`

Short Descriptions

<code>4channels</code>	generates four TIFF files from RAW data, with one file per channel
<code>ddraw_half</code>	emulates running "ddraw -h" (see DCRAW manpage)
<code>mem_image</code>	emulates running "ddraw [-4] [-6] [-e]" (see DCRAW manpage)
<code>postprocessing_benchmark</code>	creates eight different renderings from one source file. The first and fourth one should be identical
<code>simple_ddraw</code>	emulates running "ddraw [-D] [-T] [-v] [-e] [-4]" (see DCRAW manpage)
<code>ddraw_emu</code>	is an almost complete ddraw emulator (see DCRAW manpage)
<code>half_mt</code>	emulates running "ddraw -h [-w] [-a] [-v]" (see DCRAW manpage)
<code>multirender_test</code>	creates eight different renderings from one source file. The first and fourth one should be identical
<code>raw-identify</code>	emulates running "ddraw -i [-v]" (see DCRAW manpage)
<code>unprocessed_raw</code>	generates an unprocessed raw image (with masked pixels and without black subtraction)
<code>libraw.so</code>	contains functions used for parsing, editing, and saving RAW image data.

librsvg-2.58.3

Introduction to librsvg

The librsvg package contains a library and tools used to manipulate, convert and view Scalable Vector Graphic (SVG) images.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/librsvg/2.58/librsvg-2.58.3.tar.xz>
 - Download MD5 sum: 8f954b5c25ffff602b4e62813c98c8c9
 - Download size: 5.9 MB
 - Estimated disk space required: 723 MB (17 MB installed), add 545 MB for tests
 - Estimated build time: 0.7 SBU (add 0.7 SBU for tests; both using parallelism=4)

librsvg Dependencies

Required

[Cairo-1.18.0](#), [gdk-pixbuf-2.42.12](#), [Pango-1.54.0](#), and [rustc-1.80.1](#)

Note

An Internet connection is needed for building this package.

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[docutils-0.21.2](#) (for man pages), [Gi-DocGen-2024.1](#) (for documentation), and [Xorg_Fonts](#) (for tests)

Installation of librsvg

Install librsvg by running the following commands:

```
./configure --prefix=/usr      \
            --enable-vala    \
            --disable-static \
            --docdir=/usr/share/doc/librsvg-2.58.3 &&
make
```

To test the results, issue:

```
cargo update --precise 0.3.36 time &&  
LC_ALL=C make check -k
```

All 68 tests should pass.

Now, as the *root* user:

```
make DOC_INSTALL_DIR='$(docdir)' install
```

Note

If you installed the package on to your system using a "DESTDIR" method, an important file was not installed and should be copied and/or generated. Generate it using the following command as the `root` user:

```
gdk-pixbuf-query-loaders --update-cache
```

Command Explanations

`--enable-vala`: This switch enables building of the Vala bindings. Remove this switch if you don't have [Vala-0.56.17](#) installed.

--disable-static: This switch prevents installation of static versions of the libraries.

`DOC_INSTALL_DIR='$(docdir)'`: This override ensures installing the API documentation into the expected location if [Gi-DocGen-2024.1](#) is installed.

`cargo update --precise=0.3.36 time`: This updates the `Cargo.lock` file to refer to version 0.3.36 of the time crate for the test suite. Originally it refers to version 0.3.34, which fails to build with Rustc-1.80.0 or later.

`--disable-introspection`: Use this switch if you have not installed Gobject Introspection.

`--disable-gtk-doc`: This switch prevents building the API documentation, even if [Gi-DocGen-2024.1](#) (despite the name of the option) is available.

Contents

Installed Programs: `rsvg-convert`

Installed Library: `librsvg-2.so` and `libpixbufloader-svg.so` (installed in `/usr/lib/gdk-pixbuf-2.0/2.10.0/loaders`)

Installed Directories: `/usr/include/librsvg-2.0` and `/usr/share/doc/librsvg-2.58.3`

Short Descriptions

<code>rsvg-convert</code>	is used to convert images into PNG, PDF, PS, SVG and other formats
<code>librsvg-2.so</code>	provides the functions to render Scalable Vector Graphics
<code>libpixbufloader-svg.so</code>	is the Gdk Pixbuf plugin that allows GTK+ applications to render Scalable Vector Graphics images

Libspiro-20220722

Introduction to libspiro

Libspiro will take an array of spiro control points and convert them into a series of bezier splines which can then be used in the myriad of ways the world has come to use beziers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/fontforge/libspiro/releases/download/20220722/libspiro-dist-20220722.tar.gz>
- Download MD5 sum: c21f86e6c1ad65ed4cb1f754f6d7563c
- Download size: 428 KB
- Estimated disk space required: 5.1 MB (add 1.3 MB if running the tests)
- Estimated build time: 0.1 SBU

Installation of libspiro

Install libspiro by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: `libspiro.so`

Installed Directories: None

Short Descriptions

`libspiro.so` is a shareable library that can be used by programs to do the Spiro computations for you

libtiff-4.6.0

Introduction to libtiff

The libtiff package contains the TIFF libraries and associated utilities. The libraries are used by many programs for reading and writing TIFF files and the utilities are used for general work with TIFF files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.osgeo.org/libtiff/tiff-4.6.0.tar.gz>
- Download MD5 sum: fc7d49a9348b890b29f91a4ecadd5b49
- Download size: 3.4 MB
- Estimated disk space required: 46 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libtiff Dependencies

Recommended

[CMake-3.30.2](#)

Optional

[Freeglut-3.6.0](#) (required for `tiffgt`), [libjpeg-turbo-3.0.1](#), [sphinx-8.0.2](#), [libwebp-1.4.0](#), [JBIG-KIT](#), and [LERC](#)

Installation of libtiff

Install libtiff by running the following commands:

```
mkdir -p libtiff-build &&
cd      libtiff-build &&

cmake -D CMAKE_INSTALL_DOCDIR=/usr/share/doc/libtiff-4.6.0 \
-D CMAKE_INSTALL_PREFIX=/usr -G Ninja .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: `tifffcp`, `tiffdump`, `tiffinfo`, `tiffset`, and `tiffsplits`

Installed Libraries: `libtiff.so` and `libtiffxx.so`

Installed Directory: `/usr/share/doc/tiff-4.6.0`

Short Descriptions

<code>tifffcp</code>	copies (and possibly converts) a TIFF file
<code>tiffdump</code>	prints verbatim information about TIFF files
<code>tiffinfo</code>	prints information about TIFF files
<code>tiffset</code>	sets the value of a TIFF header to a specified value
<code>tiffsplits</code>	splits a multi-image TIFF into single-image TIFF files
<code>libtiff.so</code>	contains the API functions used by the libtiff programs as well as other programs to read and write TIFF files

`libtiffxx.so` contains the C++ API functions used by programs to read and write TIFF files

libwebp-1.4.0

Introduction to libwebp

The libwebp package contains a library and support programs to encode and decode images in WebP format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://storage.googleapis.com/downloads.webmproject.org/releases/webp/libwebp-1.4.0.tar.gz>
- Download MD5 sum: 07daaa13bef03b7be07f11977b159cb8
- Download size: 4.0 MB
- Estimated disk space required: 45 MB
- Estimated build time: 0.1 SBU (with parallelism=4)

libwebp Dependencies

Recommended

[libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), and [sdl12-compat-1.2.68](#) (for improved 3D Acceleration)

Optional

[Freeglut-3.6.0](#) and [giflib-5.2.2](#)

Installation of libwebp

Install libwebp by running the following commands:

```
./configure --prefix=/usr      \
--enable-libwebpmux          \
--enable-libwebpdemux        \
--enable-libwebpdecoder      \
--enable-libwebpextras       \
--enable-swap-16bit-csp     \
--disable-static             &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-swap-16bit-csp`: This switch enables byte swap for 16 bit colorspaces.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: cwebp, dwebp, gif2webp, img2webp, vwebp, webpinfo, and webpmux

Installed Library: libsharpyuv.so, libwebpdecoder.so, libwebpdemux.so, libwebpmux.so, and libwebp.so

Installed Directory: /usr/include/webp

Short Descriptions

<code>cwebp</code>	compresses an image using the WebP format
<code>dwebp</code>	decompresses WebP files into PNG, PAM, PPM or PGM images

<code>gif2webp</code>	converts a GIF image to a WebP image
<code>img2webp</code>	creates an animated WebP file from a sequence of input images
<code>vwebp</code>	decompresses a WebP file and displays it in a window
<code>webpinfo</code>	prints out the chunk level structure of WebP files along with performing basic integrity checks
<code>webpmux</code>	creates animated WebP files from non-animated WebP images, extracts frames from animated WebP images, and manages XMP/EXIF metadata and the ICC profile
<code>libwebp.so</code>	contains the API functions for WebP encoding and decoding

mypaint-brushes-1.3.1

Introduction to mypaint-brushes

The mypaint-brushes package contains brushes used by packages which use libmypaint.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mypaint/mypaint-brushes/releases/download/v1.3.1/mypaint-brushes-1.3.1.tar.xz>
- Download MD5 sum: 7241032d814cb91d2baae7d009a2a2e0
- Download size: 1.3 MB
- Estimated disk space required: 3.4 MB
- Estimated build time: less than 0.1 SBU

mypaint-brushes Dependencies

Required at runtime

[libmypaint-1.6.1](#)

Installation of mypaint-brushes

Install mypaint-brushes by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None.

Installed Libraries: None.

Installed Directories: /usr/share/mypaint-data.

newt-0.52.24

Introduction to newt

Newt is a programming library for color text mode, widget based user interfaces. It can be used to add stacked windows, entry widgets, checkboxes, radio buttons, labels, plain text fields, scrollbars, etc., to text mode user interfaces. Newt is based on the S-Lang library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://releases.pagure.org/newt/newt-0.52.24.tar.gz>

- Download MD5 sum: 9a0630e2f59eaa3037aec94989c36c4a
- Download size: 176 KB
- Estimated disk space required: 3.0 MB
- Estimated build time: less than 0.1 SBU

Newt Dependencies

Required

[popt-1.19](#) and [slang-2.3.3](#)

Recommended

[GPM-1.20.7](#) (runtime)

Installation of newt

Install newt by running the following command:

```
sed -e '/install -m 644 $(LIBNEWT)/ s/^/#/' \
-e '$(LIBNEWT) : ./rv/ s/^/#/' \
-e 's/$ $(LIBNEWT) /$(LIBNEWTSH) /g' \
-i Makefile.in &&

./configure --prefix=/usr \
--with-gpm-support \
--with-python=python3.12 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed -e ... -i Makefile.in`: Disables installation of a static library.

`--with-gpm-support`: This switch enables mouse support for newt applications through GPM.

`--with-python=python3.12`: By giving explicitly the name of the directory where python modules reside, this switch prevents building the python2 module.

Contents

Installed Programs: whiptail

Installed Library: libnewt.so, whiptcl.so, and /usr/lib/python3.12/site-packages/_snack.so

Installed Directories: None

Short Descriptions

<code>whiptail</code>	displays dialog boxes from shell scripts
<code>libnewt.so</code>	is the library for color text mode, widget based user interfaces

opencv-4.10.0

Introduction to opencv

The opencv package contains graphics libraries mainly aimed at real-time computer vision.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/opencv/opencv/archive/4.10.0/opencv-4.10.0.tar.gz>
- Download MD5 sum: adaf23e87339e6df6d50d68001138ccc
- Download size: 91 MB
- Estimated disk space required: 698 MB (with opencv-contrib)
- Estimated build time: 3.0 SBU (using parallelism=8)

Additional Downloads

- Optional additional modules: https://github.com/opencv/opencv_contrib/archive/4.10.0/opencv_contrib-4.10.0.tar.gz
- Optional additional modules md5sum: 5a4bc09177b74813c72648c117783dba
- Optional additional modules Size: 53 MB

Note

One additional file that starts with "ippicv" (integrated performance primitives) will be automatically downloaded during the cmake portion of the build procedure. This download is specific to the system architecture.

opencv Dependencies

Required

[CMake-3.30.2](#) and [UnZip-6.0](#)

Recommended

[FFmpeg-7.0.2](#), [gst-plugins-base-1.24.7](#), [GTK+-3.24.43](#), [JasPer-4.2.4](#), [libexif-0.6.24](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [OpenJPEG-2.5.2](#), [v4l-utils-1.28.1](#), and [xine-lib-1.2.13](#)

Optional

[apache-ant-1.10.14](#), [Doxygen-1.12.0](#), [Java-22.0.2](#), [NumPy-2.1.0](#), [Protobuf-27.3](#), [ATLAS](#), [blas](#), [Cuda](#), [Eigen](#), [OpenEXR](#), [GDAL](#), [lapack](#), [libdc1394](#), [Threading Building Blocks \(TBB\)](#), and [VTK - The Visualization Toolkit](#),

Installation of opencv

If you downloaded the optional modules, unpack them now:

```
tar -xf ../opencv_contrib-4.10.0.tar.gz
```

Install opencv by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
      -D CMAKE_BUILD_TYPE=Release      \
      -D ENABLE_CXX11=ON              \
      -D BUILD_PERF_TESTS=OFF        \
      -D WITH_XINE=ON                \
      -D BUILD_TESTS=OFF             \
      -D ENABLE_PRECOMPILED_HEADERS=OFF \
      -D CMAKE_SKIP_INSTALL_RPATH=ON   \
      -D BUILD_WITH_DEBUG_INFO=OFF    \
      -W no-dev ..                  &&
make
```

The package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D WITH_XINE=ON`: This option instructs the make procedure to use [xine-lib-1.2.13](#).

`-D ENABLE_PRECOMPILED_HEADERS=OFF`: This option is needed for compatibility with gcc-6.1 and later.

`-D OPENCV_EXTRA_MODULES_PATH=../opencv_contrib-4.10.0/modules`: instructs the build system to build additional modules.

Contents

Installed Programs: opencv_annotation, opencv_interactive-calibration, opencv_model_diagnostics, opencv_version, opencv_visualisation, and setup_vars_opencv4.sh

Installed Libraries: libopencv_calib3d.so, libopencv_core.so, libopencv_dnn.so, libopencv_features2d.so, libopencv_flann.so, libopencv_gapi.so, libopencv_highgui.so, libopencv_imgcodecs.so, libopencv_imgproc.so, libopencv_ml.so, libopencv_objdetect.so, libopencv_photo.so, libopencv_stitching.so, libopencv_video.so, and libopencv_videoio.so

Installed Directories: /usr/include/opencv4, /usr/lib/cmake/opencv4, /usr/lib/python3.12/site-packages/cv2, /usr/share/licenses/opencv4, /usr/share/opencv4, and /usr/share/java/opencv4

OpenJPEG-2.5.2

Introduction to OpenJPEG

OpenJPEG is an open-source implementation of the JPEG-2000 standard. OpenJPEG fully respects the JPEG-2000 specifications and can compress/decompress lossless 16-bit images.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/uclouvain/openjpeg/archive/v2.5.2/openjpeg-2.5.2.tar.gz>
- Download MD5 sum: f9ee64845881a15109ed0aa73a12202f
- Download size: 1.8 MB
- Estimated disk space required: 13 MB (add 1.7 GB for tests)
- Estimated build time: 0.2 SBU (add 0.8 SBU for tests)

OpenJPEG Dependencies

Required

[CMake-3.30.2](#)

Optional

[git-2.46.0](#) (for tests), [Little CMS-2.16](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), and [Doxygen-1.12.0](#) (to build the API documentation)

Installation of OpenJPEG

Install OpenJPEG by running the following commands:

```
mkdir -v build &&
cd      build &&

cmake -D CMAKE_BUILD_TYPE=Release \
      -D CMAKE_INSTALL_PREFIX=/usr \
      -D BUILD_STATIC_LIBS=OFF .. &&
make
```

If you wish to run the tests, some additional files are required. Download these files and run the tests using the following commands, but note that 8 tests are known to fail:

```
git clone https://github.com/uclouvain/openjpeg-data.git --depth 1 &&
OPJ_DATA_ROOT=$PWD/openjpeg-data cmake -D BUILD_TESTING=ON .. &&
```

```
make  
make test
```

&&

Now, as the `root` user:

```
make install &&  
cp -rv ./doc/man -T /usr/share/man
```

Contents

Installed Programs: `opj_compress`, `opj_decompress`, and `opj_dump`

Installed Libraries: `libopenjp2.so`

Installed Directories: `/usr/include/openjpeg-2.5` and `/usr/lib/openjpeg-2.5`

Short Descriptions

<code>opj_compress</code>	converts various image formats to the jpeg2000 format
<code>opj_decompress</code>	converts jpeg2000 images to other image types
<code>opj_dump</code>	reads in a jpeg2000 image and dumps the contents to stdout

Pixman-0.43.4

Introduction to Pixman

The Pixman package contains a library that provides low-level pixel manipulation features such as image compositing and trapezoid rasterization.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cairographics.org/releases/pixman-0.43.4.tar.gz>
- Download MD5 sum: 21b18058dea39ad48f32d3199b8ffe40
- Download size: 776 KB
- Estimated disk space required: 27 MB (With tests)
- Estimated build time: 0.2 SBU (Using parallelism=4; with tests)

Pixman Dependencies

Optional

[libpng-1.6.43](#) and [GTK+-2](#) (for tests and demos)

Installation of Pixman

Install Pixman by running the following commands:

```
mkdir build &&  
cd build &&  
  
meson setup --prefix=/usr --buildtype=release .. &&  
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Library: libpixman-1.so

Installed Directory: /usr/include/pixman-1

Short Descriptions

libpixman-1.so contains functions that provide low-level pixel manipulation features

Poppler-24.08.0

Introduction to Poppler

The Poppler package contains a PDF rendering library and command line tools used to manipulate PDF files. This is useful for providing PDF rendering functionality as a shared library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://poppler.freedesktop.org/poppler-24.08.0.tar.xz>
- Download MD5 sum: 5edd19a7ef270793374a3a5599bf872f
- Download size: 1.8 MB
- Estimated disk space required: 76 MB (with Qt6 library; add 13 MB for tests)
- Estimated build time: 0.9 SBU (with parallelism=4, tests, and Qt6 library)

Additional Downloads

Poppler Encoding Data

- Download (HTTP): <https://poppler.freedesktop.org/poppler-data-0.4.12.tar.gz>
- Download MD5 sum: 67ee4a40aa830b1f6e2560ce5f6471ba
- Download size: 4.3 MB
- Estimated disk space required: 26 MB
- Estimated build time: less than 0.1 SBU

The additional package consists of encoding files for use with Poppler. The encoding files are optional and Poppler will automatically read them if they are present. When installed, they enable Poppler to render CJK and Cyrillic properly.

Poppler Dependencies

Required

[CMake-3.30.2](#), [Fontconfig-2.15.0](#), and [GLib-2.80.4](#) (with GObject Introspection)

Recommended

[Boost-1.86.0](#), [Cairo-1.18.0](#), [GPGME-1.23.2](#), [Little CMS-2.16](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), [nss-3.103](#), [OpenJPEG-2.5.2](#), and [Qt-6.7.2](#) (required for PDF support in [okular-24.08.0](#))

Optional

[curl-8.9.1](#), [gdk-pixbuf-2.42.12](#), [git-2.46.0](#) (for downloading test files), [GTK-Doc-1.34.0](#), [GTK+-3.24.43](#), and [qt5-components-5.15.14](#)

Installation of Poppler

Now, install Poppler by running the following commands:

```
mkdir build &&
cd build &&
cmake -D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_PREFIX=/usr \
```

```

-D TESTDATADIR=$PWD/testfiles \
-D ENABLE_QT5=OFF \
-D ENABLE_UNSTABLE_API_ABI_HEADERS=ON \
-G Ninja ... &&
ninja

```

In order to run the test suite, some testcases are needed and can be obtained only from a git repository. The command to download them is: `git clone --depth 1 https://gitlab.freedesktop.org/poppler/test.git testfiles`. Then issue: `LC_ALL=en_US.UTF-8 ninja test`. One subtest of `check_qt6_signature_basics` is known to fail.

Now, as the `root` user:

```
ninja install
```

To install the documentation, run the following commands as `root`:

```
install -v -m755 -d /usr/share/doc/poppler-24.08.0 &&
cp -vr ../../glib/reference/html /usr/share/doc/poppler-24.08.0
```

Poppler Data

If you downloaded the additional encoding data package, install it by issuing the following commands:

```
tar -xf ../../poppler-data-0.4.12.tar.gz &&
cd poppler-data-0.4.12
```

Now, as the `root` user:

```
make prefix=/usr install
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply a higher level of compiler optimizations.

`-D TESTDATADIR=$PWD/testfiles`: Tells the test programs where the auxiliary files are located.

`-D ENABLE_QT5=OFF`: This switch is needed to prevent an error when Qt5 is not installed. Remove it if you have installed [Qt5-components-5.15.14](#).

`-D ENABLE_UNSTABLE_API_ABI_HEADERS=ON`: Installs some old Xpdf headers required by certain programs.

`-D ENABLE_GTK_DOC=ON`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

`-D ENABLE_QT6=OFF`: Use this parameter if [Qt-6.7.2](#) is not installed.

`-D ENABLE_BOOST=OFF`: Use this parameter if you have not installed boost (the Splash backend for Qt5 recommends boost).

`-D ENABLE_NSS3=OFF`: Use this parameter if you have not installed nss.

`-D ENABLE_GPGME=OFF`: Use this parameter if you have not installed gpgme.

`-D ENABLE_LIBTIFF=OFF`: Use this parameter if you have not installed libtiff.

`LC_ALL=en_US.UTF-8 ninja test`: Runs the test suite. The environment variable `LC_ALL=en_US.UTF-8` is only needed if the default locale does not include UTF-8.

Contents

Installed Programs: pdfattach, pdfdetach, pdffonts, pdfimages, pdfinfo, pdfseparate, pdfsig, pdftocairo, pdftohtml, pdftoppm, pdftops, pdftotext, and pdfunite

Installed Libraries: libpoppler.so, libpoppler-cpp.so, libpoppler-glib.so, libpoppler-qt5.so, and (optionally) libpoppler-qt6.so

Installed Directories: /usr/include/poppler, /usr/share/poppler, and /usr/share/doc/poppler-24.08.0

Short Descriptions

<code>pdfattach</code>	adds a new embedded file to an existing PDF file
<code>pdfdetach</code>	lists or extracts embedded files from PDF files
<code>pdffonts</code>	lists the fonts used in a PDF file along with various information for each font
<code>pdfimages</code>	saves images from a PDF file as PPM, PBM, or JPEG files

<code>pdfinfo</code>	prints the contents of the 'Info' dictionary (plus some other useful information) from a PDF file
<code>pdfseparate</code>	extracts single pages from a PDF file
<code>pdfsig</code>	verifies the digital signatures in a PDF document
<code>pdftocairo</code>	converts a PDF file to one of several formats (PNG, JPEG, PDF, PS, EPS, SVG) using the cairo output device of the poppler library
<code>pdftohtml</code>	converts a PDF file to HTML
<code>pdftoppm</code>	converts PDF files to PBM, PGM and PPM formats
<code>pdftops</code>	converts PDF files to Postscript format
<code>pdftotext</code>	converts PDF files to plain text
<code>pdfunite</code>	merges several PDF files, in the order of their occurrence on the command line, to one PDF output file
<code>libpoppler.so</code>	contains the API functions to render PDF files
<code>libpoppler-cpp.so</code>	is a C++ backend for rendering PDF files
<code>libpoppler-glib.so</code>	is a wrapper library used to interface the PDF rendering functions with GTK+
<code>libpoppler-qt5.so</code>	is a wrapper library used to interface the PDF rendering functions with Qt 5
<code>libpoppler-qt6.so</code>	is a wrapper library used to interface the PDF rendering functions with Qt 6

Potrace-1.16

Introduction to Potrace

Potrace™ is a tool for transforming a bitmap (PBM, PGM, PPM, or BMP format) into one of several vector file formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/potrace/potrace-1.16.tar.gz>
- Download MD5 sum: 5f0bd87ddd9a620b0c4e65652ef93d69
- Download size: 644 KB
- Estimated disk space required: 7.1 MB (including tests)
- Estimated build time: less than 0.1 SBU (including tests)

Potrace Dependencies

Recommended

[LLVM-18.1.7](#) (including `clang`)

Installation of Potrace

Install Potrace by running the following commands:

```
./configure --prefix=/usr \
           --disable-static \
           --docdir=/usr/share/doc/potrace-1.16 \
           --enable-a4 \
           --enable-metric \
           --with-libpotrace \
make
```

To run the test suite, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

--enable-a4: Use A4 as the default paper size.
--enable-metric: Use metric units (centimeters) as default
--disable-static: This switch prevents installation of static versions of the libraries.
--with-libpotrace: Install the library and headers.

Contents

Installed Programs: mbitmap, potrace

Installed Libraries: libpotrace.so

Installed Directories: /usr/share/doc/potrace-1.16

Short Descriptions

mbitmap	transforms images into bitmaps with scaling and filtering
potrace	transforms bitmaps into vector graphics
libpotrace.so	is a library for transforming bitmaps into vector graphics

Qpdf-11.9.1

Introduction to Qpdf

The Qpdf package contains command-line programs and a library that does structural, content-preserving transformations on PDF files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/qpdf/qpdf/releases/download/v11.9.1/qpdf-11.9.1.tar.gz>
- Download MD5 sum: 22dcfec7700e4d8b08a116ecc1c529ac
- Download size: 18 MB
- Estimated disk space required: 272 MB (add 1 MB for tests)
- Estimated build time: 0.6 SBU (using parallelism=4; add 0.5 SBU for tests)

Qpdf Dependencies

Required

[libjpeg-turbo-3.0.1](#)

Optional

[ghostscript-10.03.1](#), [GnuTLS-3.8.7.1](#), [libtiff-4.6.0](#), [sphinx-8.0.2](#) with [sphinx_rtd_theme-2.0.0](#), and [texlive-20240312](#) or [install-tl-unx](#)

Installation of Qpdf

Install Qpdf by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_STATIC_LIBS=OFF \
-D CMAKE_INSTALL_DOCDIR=/usr/share/doc/qpdf-11.9.1 \
.. &&
make
```

To test the results, issue: `ctest`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: fix-qdf, qpdf, and zlib-flate

Installed Library: libqpdf.so

Installed Directories: /usr/lib/cmake/qpdf, /usr/include/qpdf, and /usr/share/doc/qpdf-11.9.1

Short Descriptions

<code>fix-qdf</code>	is used to repair PDF files in QDF form after editing
<code>qpdf</code>	is used to convert one PDF file to another equivalent PDF file
<code>zlib-flate</code>	is a raw zlib compression program
<code>libqpdf.so</code>	contains the Qpdf API functions

qrencode-4.1.1

Introduction to qrencode

Qrencode is a fast and compact library for encoding data in a QR Code symbol, a 2D symbology that can be scanned by handheld terminals such as a mobile phone with a CCD sensor.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://fukuchi.org/works/qrencode/qrencode-4.1.1.tar.bz2>
- Download MD5 sum: de7185bcab635a34730e1b73d4efa705
- Download size: 451 KB
- Estimated disk space required: 5.0 MB (with documentation, add 5 MB for tests)
- Estimated build time: less than 0.1 SBU, add 0.1 SBU for tests

Qrencode Dependencies

Recommended

[libpng-1.6.43](#)

Optional

[Doxygen-1.12.0](#) for generating documentation, and [SDL2-2.30.6](#) for tests

Installation of qrencode

Install libqrencode by running the following commands:

```
./configure --prefix=/usr &&  
make
```

If you have installed [Doxygen-1.12.0](#), you can build the documentation by issuing:

```
doxygen
```

The tests must be run after installing the package.

Now, as the `root` user:

```
make install
```

If you have built the optional documentation, install it as the `root` user:

```
install -vdm 755 /usr/share/doc/qrencode-4.1.1 &&  
mv html/* /usr/share/doc/qrencode-4.1.1
```

To test the results, if you have passed the `--with-tests` option to `configure`, issue: `make check`.

Command Explanations

`--with-tests`: This option allows building the test programs. It requires [SDL2-2.30.6](#).

`--without-tools`: This option prevents building the `qrencode` executable, removing the need for [libpng-1.6.43](#).

Contents

Installed Program: qrencode

Installed Library: libqrencode.so

Installed Directory: /usr/share/doc/qrcode-4.1.1 (optional)

Short Descriptions

<code>qrencode</code>	encodes input data in a QR Code and saves it as a PNG or EPS image
<code>libqrencode.so</code>	contains functions for encoding data in a QR code symbol

sassc-3.6.2

Introduction to sassc

SassC is a wrapper around libsass, a CSS pre-processor language.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/sass/sassc/archive/3.6.2/sassc-3.6.2.tar.gz>
- Download MD5 sum: 4c3b06ce2979f2a9f0a35093e501d8bb
- Download size: 28 KB
- Estimated disk space required: 5.1 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Download (HTTP): <https://github.com/sass/libsass/archive/3.6.6/libsass-3.6.6.tar.gz>
- Download MD5 sum: afda97284d75a030cabadf5b9f998a3b
- Download size: 336 KB
- Estimated disk space required: 135 MB
- Estimated build time: 0.4 SBU (Using parallelism=4)

Installation of sassc

First, build the library:

```
tar -xf ../libsass-3.6.6.tar.gz &&
pushd libsass-3.6.6 &&

autoreconf -fi &&
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Build the command line wrapper:

```
popd &&
autoreconf -fi &&

./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: sassc

Installed Libraries: libsass.so

Installed Directories: /usr/include/sass

Short Descriptions

`sassc` provides a command line interface to the libsass library

webp-pixbuf-loader-0.2.7

Introduction to webp-pixbuf-loader

The webp-pixbuf-loader package contains a library that allows gdk-pixbuf to load and process webp images.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/aruiz/webp-pixbuf-loader/archive/0.2.7/webp-pixbuf-loader-0.2.7.tar.gz>
- Download MD5 sum: e97025dc70178877dbd041776f151947
- Download size: 8.8 MB
- Estimated disk space required: 9.7 MB
- Estimated build time: less than 0.1 SBU (with tests)

webp-pixbuf-loader Dependencies

Required

[gdk-pixbuf-2.42.12](#) and [libwebp-1.4.0](#)

Installation of webp-pixbuf-loader

Install webp-pixbuf-loader by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

The webp format needs to be added to the loaders cache:

```
gdk-pixbuf-query-loaders --update-cache
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`gdk-pixbuf-query-loaders --update-cache`: This command updates the gdk-pixbuf loader cache so that it knows the webp loader has been installed.

Contents

Installed Programs: None

Installed Libraries: libpixbufloader-webp.so

Installed Directories: None

Short Descriptions

`libpixbufloader-webp.so` contains functions that allow gdk-pixbuf to load webp images

woff2-1.0.2

Introduction to WOFF2

WOFF2 is a library for converting fonts from the TTF format to the WOFF 2.0 format. It also allows decompression from WOFF 2.0 to TTF. The WOFF 2.0 format uses the Brotli compression algorithm to compress fonts suitable for downloading in CSS @font-face rules.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/google/woff2/archive/v1.0.2/woff2-1.0.2.tar.gz>
- Download MD5 sum: 793c8844845351cb80730a74937e411b
- Download size: 39 KB
- Estimated disk space required: 1.6 MB
- Estimated build time: less than 0.1 SBU

WOFF2 Dependencies

Required

[Brotli-1.1.0](#) and [CMake-3.30.2](#)

Installation of WOFF2

Install WOFF2 by running the following commands:

```
mkdir out          &&
cd out           &&
cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D CMAKE_SKIP_INSTALL_RPATH=ON ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

Contents

Installed Program: None

Installed Libraries: libwoff2common.so, libwoff2dec.so and libwoff2enc.so

Installed Directory: /usr/include/woff2

Short Descriptions

libwoff2common.so	provides shared data used by the libwoff2dec and libwoff2enc libraries
libwoff2dec.so	is the WOFF2 decoder library
libwoff2enc.so	is the WOFF2 encoder library

Chapter 11. General Utilities

This chapter contains various utilities that do not fit conveniently into other chapters. Programs include some documentation generators, several utilities for manipulating text and graphics, programs for listing files, a program for entering PIN numbers and pass-phrases, and a connection manager.

Asciidoctor-2.0.23

Introduction to Asciidoctor

Asciidoctor is a fast, open source text processor and publishing toolchain for converting AsciiDoc content to HTML5, DocBook, PDF, and other formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/asciidoctor/asciidoctor/archive/v2.0.23/asciidoctor-2.0.23.tar.gz>
- Download MD5 sum: e629ec9b6a82a21b4afaab7d5d495e52
- Download size: 1.6 MB
- Estimated disk space required: 5.5 MB
- Estimated build time: less than 0.1 SBU

Asciidoctor Dependencies

Required

[Ruby-3.3.4](#)

Installation of Asciidoctor

Note

When building this package, the following message may appear:

```
fatal: not a git repository (or any of the parent directories): .git
```

. This is normal, and the package will continue building past this point.

Build the Ruby gem:

```
gem build asciidoctor.gemspec
```

The test suite needs many Ruby gems beyond the scope of BLFS.

Now, as the `root` user:

```
gem install asciidoctor-2.0.23.gem &&
install -vM644 man/asciidoctor.1 /usr/share/man/man1
```

Contents

Installed Programs: asciidoctor

Installed Libraries: None

Installed Directories: /usr/lib/ruby/gems/3.3.0/gems/asciidoc-2.0.23 and /usr/lib/ruby/gems/3.3.0/doc/asciidoc-2.0.23

Short Descriptions

`asciidoc` converts AsciiDoc source files to HTML, DocBook, and other formats

Bogofilter-1.2.5

Introduction to Bogofilter

The Bogofilter application is a mail filter that classifies mail as spam or ham (non-spam) by a statistical analysis of the message's header and content (body).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/bogofilter/bogofilter-1.2.5.tar.xz>
- Download MD5 sum: 8763f87adfff7b802ced177d8c654539
- Download size: 784 KB
- Estimated disk space required: 19 MB
- Estimated build time: 0.1 SBU

Bogofilter Dependencies

Recommended

[gsl-2.8](#), [libxml2-2.13.3](#), and [SQLite-3.46.1](#)

Optional

[Imdb-0.9.31](#), [xmlto-0.0.29](#), [Berkeley DB](#) (deprecated), [QDBM](#) and [TokyoCabinet](#)

Note

If you do not install the recommended [gsl-2.8](#) package then a statically linked shipped version will be used instead.

Installation of Bogofilter

Note

If you plan to change the version of your database library on an existing installation, or to change to a different database, read the warning at the top of the RELEASE.NOTES file.

Install Bogofilter by running the following commands:

```
./configure --prefix=/usr \
            \
            --sysconfdir=/etc/bogofilter \
            \
            --with-database=sqlite3      &&
make
```

To test the results, issue `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

--with-database=sqlite3: This switch makes Bogofilter use [SQLite-3.46.1](#) as the database, instead of Berkeley DB.
--with-database={lmdb,qdbm,tokycabinet}: This switch also allows to use lmdb, qdbm, or tokycabinet as the database.

Contents

Installed Programs: bf_compact, bf_copy, bf_tar, bogofilter, bogolexer, bogotune, bogoupgrade and bogoutil
Installed Libraries: None
Installed Directories: /etc/bogofilter

Short Descriptions

bf_compact	creates a more compact bogofilter working directory with a dump/load cycle
bf_copy	copies a bogofilter working directory to another directory
bf_tar	bundles a bogofilter working directory in tar format and copies it to standard output
bogofilter	is a fast Bayesian spam filter
bogolexer	is used to separate messages into tokens and to test new versions of the lexer.l code
bogotune	attempts to find optimum parameter settings for bogofilter
bogoupgrade	upgrades the bogofilter database to the current version
bogoutil	dumps, loads, and maintains bogofilter database files

Compface-1.5.2

Introduction to Compface

Compface provides utilities and a library to convert from/to X-Face format, a 48x48 bitmap format used to carry thumbnails of email authors in a mail header.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/compface/compface-1.5.2.tar.gz>
- Download MD5 sum: 62f4f79c0861ad292ba3cf77b4c48319
- Download size: 47 KB
- Estimated disk space required: 520 KB
- Estimated build time: less than 0.1 SBU

Installation of Compface

First, fix the configure script so that it's C99 compatible:

```
autoreconf
```

Next, fix a build failure that occurs with GCC 14:

```
sed -e '/compface.h/a #include <unistd.h>' \
-i cmain.c \
-i uncmain.c
```

Install Compface by running the following commands:

```
./configure --prefix=/usr --mandir=/usr/share/man &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -m755 -v xbm2xface.pl /usr/bin
```

Contents

Installed Programs: compface, uncompface and xbm2xface.pl

Installed Library: libcompface.a

Installed Directories: None

Short Descriptions

<code>compface</code>	is a filter for generating highly compressed representations of 48x48x1 face image files
<code>uncompface</code>	is an inverse filter which performs an inverse transformation with no loss of data
<code>xbm2xface.pl</code>	is a script to generate xfaces
<code>libcompface.a</code>	allows the compression and decompression algorithms to be used in other programs such as MTAs

desktop-file-utils-0.27

Introduction to Desktop File Utils

The Desktop File Utils package contains command line utilities for working with [Desktop entries](#). These utilities are used by Desktop Environments and other applications to manipulate the MIME-types application databases and help adhere to the Desktop Entry Specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/desktop-file-utils/releases/desktop-file-utils-0.27.tar.xz>
- Download MD5 sum: fdf9215aea91acb6aebc43f770537f2c
- Download size: 80 KB
- Estimated disk space required: 1.2 MB
- Estimated build time: less than 0.1 SBU

Desktop File Utils Dependencies

Required

[GLib-2.80.4](#)

Optional

[Emacs-29.4](#)

Installation of Desktop File Utils

Warning

If you are upgrading from a previous version of desktop-file-utils that used the Autotools method of installing and configuring the package, you must remove the desktop-file-edit symlink by using the following commands.

```
rm -fv /usr/bin/desktop-file-edit
```

Install Desktop File Utils by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Configuring Desktop File Utils

Configuration Information

The [XDG Base Directory](#) specification defines the standard locations for applications to place data and configuration files. These files can be used, for instance, to define the menu structure and menu items in a desktop environment.

The default location for configuration files to be installed is `/etc/xdg`, and the default locations for data files are `/usr/local/share` and `/usr/share`. These locations can be extended with the environment variables `XDG_CONFIG_DIRS` and `XDG_DATA_DIRS`, respectively. The GNOME, KDE and XFCE environments respect these settings.

When a package installs a `.desktop` file to a location in one of the base data directories, the database that maps MIME-types to available applications can be updated. For instance, the cache file at `/usr/share/applications/mimeinfo.cache` can be rebuilt by executing the following command as the `root` user:

```
install -vdm755 /usr/share/applications &&
update-desktop-database /usr/share/applications
```

Contents

Installed Programs: `desktop-file-edit`, `desktop-file-install`, `desktop-file-validate` and `update-desktop-database`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>desktop-file-edit</code>	is used to modify an existing desktop file entry
<code>desktop-file-install</code>	is used to install a new desktop file entry. It is also used to rebuild or modify the MIME-types application database
<code>desktop-file-validate</code>	is used to verify the integrity of a desktop file
<code>update-desktop-database</code>	is used to update the MIME-types application database

dos2unix-7.5.2

Introduction to dos2unix

The dos2unix package contains an any-to-any text format converter.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/dos2unix/dos2unix-7.5.2.tar.gz>
- Download MD5 sum: 646272020848c9b673de24c4e8e3422e
- Download size: 972 KB
- Estimated disk space required: 7.2 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Installation of dos2unix

Build dos2unix by running the following commands:

```
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: dos2unix, mac2unix, unix2dos, and unix2mac

Installed Libraries: None

Installed Directories: /usr/share/doc/dos2unix-7.5.2

Short Descriptions

<code>dos2unix</code>	converts plain text files in DOS format to Unix format
<code>mac2unix</code>	converts plain text files in Mac format to Unix format
<code>unix2dos</code>	converts plain text files in Unix format to DOS format
<code>unix2mac</code>	converts plain text files in Unix format to Mac format

Graphviz-12.1.0

Introduction to Graphviz

The Graphviz package contains graph visualization software. Graph visualization is a way of representing structural information as diagrams of abstract graphs and networks. Graphviz has several main graph layout programs. It also has web and interactive graphical interfaces, auxiliary tools, libraries, and language bindings.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.com/graphviz/graphviz/-/archive/12.1.0/graphviz-12.1.0.tar.bz2>
- Download MD5 sum: 2c3ed90a06a2496c67b12ef3d5f93c5a
- Download size: 25 MB
- Estimated disk space required: 217 MB
- Estimated build time: 0.7 SBU (using parallelism=4)

Graphviz Dependencies

Note

Basic usage of Graphviz does not require any libraries out of what is found in the LFS book. Its “core” rendering engine is able to generate several graphic formats, such as Postscript, SVG, VML, .fig, and Tk. Those formats can be converted to almost any other format using tools from packages such as [ImageMagick-7.1.1-36](#). The dependencies below add the ability to generate graph images in bitmap format, to display the graph image on screen, to edit a graph by viewing the resulting image directly, or to view large graphs. Since Graphviz is a dependency of several other packages in this book, it is suggested that you first build it without any dependencies, and then rebuild it when you have built enough packages to suit your needs.

Optional, for various bitmap outputs

[Pango-1.54.0](#), with [Cairo-1.18.0](#), [Xorg Libraries](#), [Fontconfig-2.15.0](#), and [libpng-1.6.43](#), to generate images in bitmap SVG, postscript, PNG, and PDF formats, or to display the image on screen. The PNG output is required for building [gegl-0.4.48](#)

Adding [GTK+-2](#) with [libjpeg-turbo-3.0.1](#) adds support for JPEG, BMP, TIF, and ICO formats, and allows displaying the image in a GTK+ window

[GD Library](#) may be used instead of Pango. It adds the ability to generate images in GIF, VRML, and GD formats, but Pango provides better outputs for the other formats, and is needed for displaying images

Other formats may be added with [libwebp-1.4.0](#) (WebP support is considered experimental), [DevIL](#), [libLASI](#), and [glitz](#)

Optional (to load graphics that may be displayed inside the nodes of a graph)

libgs.so from [ghostscript-10.03.1](#), [librsvg-2.58.3](#), and [Poppler-24.08.0](#)

Optional (to build more tools)

[Freeglut-3.6.0](#) (with [GtkGLExt](#) and [libGTS](#) for building the `smyrna` large graph viewer, which is considered experimental), and [qt5-components-5.15.14](#) (for building the `gvedit` graph editor)

Optional (to build language bindings)

[SWIG-4.2.1](#) (SWIG must be installed or no bindings will be built), [GCC-14.2.0](#) (for the go compiler), [Guile-3.0.10](#), [OpenJDK-22.0.2](#), [Lua-5.4.7](#), [PHP-8.3.10](#), [Ruby-3.3.4](#), [Tk-8.6.14](#), [Io](#), [Mono](#), [OCaml](#), and [R](#)

Optional (building tools)

[Criterion](#) (framework for tests) and [Electric Fence](#)

Optional (for building the pdf documentation)

[ghostscript-10.03.1](#) (for the `ps2pdf` command)

Installation of Graphviz

Install Graphviz by running the following commands:

```
sed -i '/LIBPOSTFIX="64"/s/64//' configure.ac &&
./autogen.sh &&
./configure --prefix=/usr \
--docdir=/usr/share/doc/graphviz-12.1.0
```

Note

A warning is generated by `autogen.sh` because the build tree is not a git repository. As a result, the build date is set to 0. To get a meaningful date in the version string, you can run:

```
sed -i "s/0/$(date +%Y%m%d)/" builddate.h
```

Whether or not you fix the date, proceed to compile the package:

```
make
```

This package does not come with a test suite that provides meaningful results.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ... configure.ac`: This command is needed to avoid installing files in `/usr/lib64`.

`--with-javaincludedir="$JAVA_HOME/include -I$JAVA_HOME/include/linux"`: If you have built [OpenJDK-22.0.2](#) in `/opt`, and you want to build the JAVA bindings, it is necessary to specify the location of the JAVA header files to configure. The configure switch is designed for only one directory, but two directories need to be included. This is possible nevertheless by using the `-I` switch inside the variable.

`--with-webp`: Even if [libwebp-1.4.0](#) is installed, it is not included in the build without this option.

`--with-smyrna`: Even if the needed dependencies are installed, the interactive graph viewer `smyrna` is not built without this option.

Configuring Graphviz

Config Files

`/usr/lib/graphviz/config`

Configuration Information

There are no specific configuration requirements for Graphviz. You may consider installing the additional plugins and tools available from the download page at <https://graphviz.org/download/source/> for additional capabilities. If additional plugins are installed, you can run `dot -c` (as the `root` user) to update the `config` file in `/usr/lib/graphviz`.

Contents

- Installed Programs:** acyclic, bcomps, ccomps, circo, cluster, dijkstra, dot, dot2gxl, dot_builtins, edgepaint, fdp, gc, gml2gv, graphml2gv, gv2gml, gv2gxl, gvcolor, gvedit, gvgen, gvmap, gvmap.sh, gvpack, gvpr, gxl2dot, gxl2gv, mm2gv, neato, nop, osage, patchwork, prune, sccmap, sfdp, tred, twopi, unflatten, and vimdot
- Installed Libraries:** libcdt.so, libcgraph.so, libgvc.so, libgvpr.so, liblab_gamut.so, libpathplan.so, libxdot.so, and several plugins in `/usr/lib/graphviz`. There are also several in subdirectories of `/usr/lib/{lua,perl5,php,python3.12,tcl8.6}`. Unfortunately, some libraries are duplicated.
- Installed Directories:** `/usr/include/graphviz`, `/usr/lib/graphviz`, `/usr/lib/tcl8.6/graphviz`, `/usr/share/doc/graphviz-12.1.0`, and `/usr/share/graphviz`

Short Descriptions

<code>acyclic</code>	is a filter that takes a directed graph as input and outputs a copy of the graph with sufficient edges reversed to make the graph acyclic
<code>bcomps</code>	decomposes graphs into their biconnected components, printing the components to standard output
<code>ccomps</code>	decomposes graphs into their connected components, printing the components to standard output
<code>circo</code>	draws graphs using a circular layout
<code>cluster</code>	takes a graph in DOT format as input, finds node clusters, and then augments the graph with this information
<code>diffimg</code>	(needs GD Library) generates an image where each pixel is the difference between the corresponding pixel in each of the two source images
<code>dijkstra</code>	reads a stream of graphs and for each computes the distance of every node from sourcenode
<code>dot</code>	draws directed graphs. It works well on DAGs and other graphs that can be drawn as hierarchies. It reads attributed graph files and writes drawings. By default, the output format dot is the input file with layout coordinates appended
<code>dot2gxl</code>	converts between graphs represented in GXL and in the DOT language. Unless a conversion type is specified using a flag, <code>gxl2dot</code> will deduce the type of conversion from the suffix of the input file, a <code>.dot</code> suffix causing a conversion from DOT to GXL, and a <code>.gxl</code> suffix causing a conversion from GXL to DOT
<code>edgepaint</code>	performs edge coloring to disambiguate crossing edges
<code>fdp</code>	draws undirected graphs using a "spring" model. It relies on a force-directed approach in the spirit of Fruchterman and Reingold
<code>gc</code>	is a graph analogue to <code>wc</code> in that it prints to standard output the number of nodes, edges, connected components or clusters contained in the input files. It also prints a total count for all graphs if more than one graph is given
<code>gml2gv</code>	converts a graph specified in the GML format to a graph in the GV (formerly DOT) format
<code>graphml2gv</code>	converts a graph specified in the GRAPHML format to a graph in the GV (formerly DOT) format
<code>gv2gml</code>	converts a graph specified in the GV format to a graph in the GML format
<code>gv2gxl</code>	converts a graph specified in the GV format to a graph in the GXL format
<code>gvcolor</code>	is a filter that sets node colors from initial seed values. Colors flow along edges from tail to head, and are averaged (as HSB vectors) at nodes
<code>gvedit</code>	provides a simple graph editor and viewer. It allows many graphs to be viewed at the same time. The text of each graph is displayed in its own text window
<code>gvgen</code>	generates a variety of simple, regularly-structured abstract graphs
<code>gvmap</code>	takes as input a graph in DOT format, finds node clusters and produces a rendering of the graph as a geographic-style map, with clusters highlighted, in xdot format
<code>gvmap.sh</code>	is a pipeline for running gvmap
<code>gvpack</code>	reads in a stream of graphs, combines the graphs into a single layout, and produces a single graph serving as the union of the input graphs
<code>gvpr</code>	is a graph stream editor inspired by <code>awk</code> . It copies input graphs to its output, possibly transforming their structure and attributes, creating new graphs, or printing arbitrary information
<code>gxl2dot</code>	converts between graphs represented in GXL and in the DOT language. Unless a conversion type is specified using a flag, <code>gxl2dot</code> will deduce the type of conversion from the suffix of the input file, a <code>.dot</code> suffix causing a conversion from DOT to GXL, and a <code>.gxl</code> suffix causing a conversion from GXL to DOT

<code>gx12gv</code>	converts between graphs represented in GXL and in the GV language
<code>mm2gv</code>	converts a sparse matrix of the Matrix Market format to a graph in the GV (formerly DOT) format
<code>neato</code>	draws undirected graphs using "spring" models. Input files must be formatted in the <code>.dot</code> attributed graph language. By default, the output of <code>neato</code> is the input graph with layout coordinates appended
<code>nop</code>	reads a stream of graphs and prints each in pretty-printed (canonical) format on stdout. If no files are given, it reads from stdin
<code>osage</code>	draws clustered graphs. It takes any graph in DOT format as input
<code>patchwork</code>	draws clustered graphs using a squarified treemap layout. It takes any graph in DOT format as input
<code>prune</code>	reads directed graphs in the same format used by <code>dot</code> and removes subgraphs rooted at nodes specified on the command line via options
<code>sccmap</code>	decomposes digraphs into strongly connected components and an auxiliary map of the relationship between components. In this map, each component is collapsed into a node. The resulting graphs are printed to stdout
<code>sfdp</code>	draws undirected graphs using the "spring" model, but it uses a multi-scale approach to produce layouts of large graphs in a reasonably short time
<code>tred</code>	computes the transitive reduction of directed graphs, and prints the resulting graphs to standard output. This removes edges implied by transitivity. Nodes and subgraphs are not otherwise affected
<code>twopi</code>	draws graphs using a radial layout. Basically, one node is chosen as the center and put at the origin. The remaining nodes are placed on a sequence of concentric circles centered about the origin, each a fixed radial distance from the previous circle
<code>unflatten</code>	is a preprocessor to <code>dot</code> that is used to improve the aspect ratio of graphs having many leaves or disconnected nodes. The usual layout for such a graph is generally very wide or tall
<code>vimdot</code>	is a simple script which launches the <code>gvim</code> or <code>vim</code> editor along with a GUI window showing the <code>dot</code> output of the edited file
<code>libcdt.so</code>	manages run-time dictionaries using standard container data types: unordered set/multiset, ordered set/multiset, list, stack, and queue
<code>libgraph.so</code>	supports graph programming by maintaining graphs in memory and reading and writing graph files. Graphs are composed of nodes, edges, and nested subgraphs
<code>libgvc.so</code>	provides a context for applications wishing to manipulate and render graphs. It provides command line parsing interfaces, common rendering code, and a plugin mechanism for renderers
<code>libpathplan.so</code>	contains functions to find the shortest path between two points in a simple polygon
<code>libxdot.so</code>	provides support for parsing and deparsing graphical operations specified by the xdot language

GTK-Doc-1.34.0

Introduction to GTK-Doc

The GTK-Doc package contains a code documenter. This is useful for extracting specially formatted comments from the code to create API documentation. This package is *optional*; if it is not installed, packages will not build the documentation. This does not mean that you will not have any documentation. If GTK-Doc is not available, the install process will copy any pre-built documentation to your system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtk-doc/1.34/gtk-doc-1.34.0.tar.xz>
- Download MD5 sum: f0e7385ba25eddb6ce0953e8cf63d1bf
- Download size: 484 KB
- Estimated disk space required: 14 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

GTK-Doc Dependencies

Required

[docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [GLib-2.80.4](#), [itstool-2.0.7](#), [libxslt-1.1.42](#), and [Pygments-2.18.0](#)

Optional

For tests: [dblatex](#) or [fop-2.9](#) (XML PDF support), [GLib-2.80.4](#), [Which-2.21](#), and Python modules [lxml-5.3.0](#), [parameterized](#), and [yelp-tools](#)

Note

The optional python modules above can be easily installed with the `pip3` command.

Installation of GTK-Doc

Install GTK-Doc by running the following commands:

```
mkdir -p build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

The test suite will hang if the package (or a previous version) is not already installed.

Now, as the `root` user:

```
ninja install
```

To test the results, issue: `ninja test`. Some tests will fail depending on optionally installed packages.

Contents

Installed Programs: gtkdocize, gtkdoc-check, gtkdoc-depscan, gtkdoc-fixxref, gtkdoc-mkdb, gtkdoc-mkhtml, gtkdoc-mkhtml2, gtkdoc-mkman, gtkdoc-mkpfd, gtkdoc-rebase, gtkdoc-scan, and gtkdoc-scangobj

Installed Libraries: None

Installed Directories: /usr/share/gtk-doc and /usr/share/cmake/GtkDoc

Short Descriptions

`gtkdoc*` these are all shell, or Python scripts used by package `Makefile` scripts to generate documentation for the package being built

Highlight-4.13

Introduction to Highlight

Highlight is an utility that converts source code to formatted text with syntax highlighting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://www.andre-simon.de/zip/highlight-4.13.tar.bz2>
- Download MD5 sum: 25e6aef8901eb5cf555f36be63ce502e
- Download size: 1.4 MB
- Estimated disk space required: 24 MB (with gui)
- Estimated build time: 0.3 SBU (Using parallelism=4; with gui)

Highlight Dependencies

Required

[Boost-1.86.0](#) and [Lua-5.4.7](#)

Optional

[Qt-6.7.2](#) (to build the GUI front-end)

Installation of Highlight

For consistency, do not compress man pages.

```
sed -i '/GZIP/s/^/#/' makefile
```

To build Highlight run the following command:

```
make
```

To build the qt6 GUI front-end, run the following command:

```
make doc_dir=/usr/share/doc/highlight-4.13/ gui
```

This package does not come with a test suite.

To install Highlight, run the following command as the `root` user:

```
make doc_dir=/usr/share/doc/highlight-4.13/ install
```

To install the GUI program, run the following command as the `root` user:

```
make install-gui
```

Command Explanations

`doc_dir=/usr/share/doc/highlight-4.13/`: installs the highlight documentation into a versioned directory. This parameter is also needed for `make gui`, because its value would be hardcoded into the gui executable. Note that the trailing "/" is necessary.

Contents

Installed Programs: `highlight` and `highlight-gui` (optional)

Installed Libraries: None

Installed Directories: /etc/highlight, /usr/share/doc/highlight-4.13, and /usr/share/highlight

Short Descriptions

`highlight` is a universal source code to formatted text converter
`highlight-gui` is the qt6 interface to `highlight`.

ibus-1.5.30

Introduction to ibus

ibus is an Intelligent Input Bus. It is a new input framework for the Linux OS. It provides a fully featured and user friendly input method user interface.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/ibus/ibus/archive/1.5.30/ibus-1.5.30.tar.gz>
- Download MD5 sum: a7facee0030e0e6adb41ecfd94be9e61
- Download size: 1.5 MB
- Estimated disk space required: 55 MB (add 3 MB for tests)
- Estimated build time: 0.4 SBU (Using parallelism=4; add 0.8 SBU for tests)

Additional Downloads

- Unicode Character Database: <https://www.unicode.org/Public/zipped/15.1.0/UCD.zip>

ibus Dependencies

Required

[DConf-0.40.0](#), [ISO Codes-4.16.0](#), and [Vala-0.56.17](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [GTK+-3.24.43](#), and [libnotify-0.8.3](#)

Optional

[GTK-4.14.5](#) (to build IM module for it), [GTK-Doc-1.34.0](#) (for generating API documentation), [D-Bus Python-1.3.2](#) and [PyGObject-3.48.2](#) (both to build the Python support library), [libxkbcommon-1.7.0](#), [Wayland-1.23.0](#) (both to build the Wayland support programs), [EmojiOne](#), and [libibusmenu](#)

Installation of ibus

First, install the Unicode Character Database as the `root` user:

```
mkdir -p /usr/share/unicode/ucd &&
unzip -o ./UCD.zip -d /usr/share/unicode/ucd
```

Fix an issue with deprecated schema entries:

```
sed -e 's@/desktop/ibus@/org/freedesktop/ibus@g' \
-i data/dconf/org.freedesktop.ibus.gschema.xml
```

If [GTK-Doc-1.34.0](#) is not installed, remove the references to it:

```
if ! [ -e /usr/bin/gtkdocize ]; then
    sed '/docs/d;/GTK_DOC/d' -i Makefile.am configure.ac
fi
```

Install ibus by running the following commands:

```
SAVE_DIST_FILES=1 NOCONFIGURE=1 ./autogen.sh &&

PYTHON=python3 \
./configure --prefix=/usr \
--sysconfdir=/etc \
--disable-python2 \
--disable-appindicator \
--disable-gtk2 \
--disable-emoji-dict &&
make
```

To test the results, issue: `make -k check`. The test named ibus-compose fails because it uses some locales not installed in LFS. The test named ibus-keypress will fail if running in a Wayland session.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-appindicator`: This switch disables use of libibusmenu. Omit if you installed the optional dependency.

`--disable-emoji-dict`: This switch disables the use of emoticon dictionaries. Omit if you installed the optional package.

`--disable-gtk2`: This switch removes the dependency on GTK+-2.

`--enable-gtk4`: This switch enables building the GTK 4 immodule. Use it if you have installed GTK 4.

`--enable-python-library`: This switch enables building the Python support library. Use it if you have installed the optional Python modules.

`--enable-wayland`: This switch enables building the Wayland support programs. It's automatically enabled if both [libxkbcommon-1.7.0](#) and [Wayland-1.23.0](#) are installed.

`NOCONFIGURE=1`: Prevent `autogen.sh` from running the generated `configure` script. We'll run the script manually instead of relying on `autogen.sh` to run it, because `autogen.sh` would set `-fsanitize=address -fsanitize=leak` in `CFLAGS` running `configure`, but those compiler options are not suitable for productive use and they may also cause a build failure.

`PYTHON=python3`: This environment variable makes the `configure` script look for Python 3. Use it if you want to build the Python 3 support library.

`SAVE_DIST_FILES=1`: This environment variable makes the `autogen.sh` script save some pre-generated header files instead of overwriting them when it is run. This prevents build failures when generating `ibusemojigen.h`.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Configuring Ibus

If GTK+-3 is installed and `--disable-gtk3` is not used, the ibus IM module for GTK+-3 will be installed. As the `root` user, update a cache file of GTK+-3 so the GTK-based applications can find the newly installed IM module and use ibus as an input method:

```
gtk-query-immodules-3.0 --update-cache
```

The command above updates the cache file for GTK+-3. GTK-4 does not require a cache file for IM modules.

Contents

Installed Programs: ibus, ibus-daemon, and ibus-setup

Installed Library: libibus-1.0.so and im-ibus.so (GTK+ Immodule)

Installed Directories: /etc/dconf/db/ibus.d, /usr/include/ibus-1.0, /usr/share/gtk-doc/html/ibus, and /usr/share/ibus

Short Descriptions

<code>ibus-daemon</code>	is the Intelligent Input Bus Daemon
<code>ibus-setup</code>	is the GTK+ program used to configure the <code>ibus-daemon</code>
<code>libibus-1.0.so</code>	contains the ibus API functions

ImageMagick-7.1.1-36

Introduction to ImageMagick

ImageMagick is a collection of tools and libraries to read, write, and manipulate an image in various image formats. Image processing operations are available from the command line. Bindings for Perl and C++ are also available.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://wwwimagemagick.org/archive/releases/ImageMagick-7.1.1-36.tar.xz>
- Download MD5 sum: 208507e5465716767ab9c73f45e8ca06
- Download size: 10 MB
- Estimated disk space required: 177 MB, 45 MB installed (with typical dependencies; add 10 MB for checks)
- Estimated build time: 0.7 SBU (with typical dependencies; add 0.7 SBU for checks, both using parallelism=4)

Note

The ImageMagick source releases are updated frequently and the version shown above may no longer be available from the download locations. You can download a more recent version and use the existing BLFS instructions to install it. Chances are that it will work just fine, but this has not been tested by the BLFS team. If the package version shown above is not available from the locations shown above, you can download it from the BLFS package server at Oregon State University:

<https://ftp.osuosl.org/pub/blfs/conglomeration/ImageMagick/>.

ImageMagick Dependencies

Recommended

[Xorg Libraries](#)

The optional dependencies listed below should be installed if you need support for the specific format or the conversion tool the dependency provides. Additional information about the dependencies can be found in the `Install-unix.txt` file located in the source tree as well as issuing the `./configure --help` command. A summary of this information, as well as some additional notes can be viewed on-line at <https://imagemagick.org/script/install-source.php>.

Optional System Utilities

Clang from [LLVM-18.1.7](#), [Cups-2.4.10](#), [cURL-8.9.1](#), [FFmpeg-7.0.2](#), [fftw-3.3.10](#), [p7zip-17.04](#) (LZMA), [SANE-1.2.1](#), [Wget-1.24.5](#), [xdg-utils-1.2.1](#), [xterm-393](#), [Dmalloc](#), [Electric Fence](#) and [PGP](#) or [GnuPG-2.4.5](#) (you'll have to do some hacking to use GnuPG), [Profiles](#)

Optional Graphics Libraries

[JasPer-4.2.4](#), [Little CMS-2.16](#), [libgxpath-0.3.2](#), [libjpeg-turbo-3.0.1](#), [libjxl-0.10.3](#), [libpng-1.6.43](#), [libraw-0.21.2](#) (for dng files, needs the thread-safe libraw_r), [librsvg-2.58.3](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [OpenJPEG-2.5.2](#), [Pango-1.54.0](#), [DjVuLibre](#), [FlashPIX \(libfpx\)](#), [FLIF](#), [JBIG-KIT](#), [libheif](#) with [libde265](#) (both needed if converting macOS heic images), [libraqm](#), [Liquid Rescale](#), [OpenEXR](#), and [RALCGM](#) (or [ralcgm](#))

Optional Graphics Utilities

[Dejavu fonts](#), [ghostscript-10.03.1](#), [Gimp-20240711](#), [Graphviz-12.1.0](#), [Inkscape-1.3.2](#), [Blender](#), [corefonts](#), [GhostPCL](#), [Gnuplot](#), [POV-Ray](#), and [Radiance](#)

Optional Conversion Tools

[Enscript-1.6.6](#), [Potrace-1.16](#), [texlive-20240312](#) (or [install-tl-unx](#)) [AutoTrace](#), [GeoExpress Command Line Utilities](#), AKA [MrSID Utilities \(binary package\)](#), [hp2xx](#), [libwmf](#), [UniConvertor](#), and [Utah Raster Toolkit](#) (or [URT-3.1b](#))

Installation of ImageMagick

Install ImageMagick by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --enable-hdri      \
            --with-modules     \
            --with-perl        \
            --disable-static   &&
make
```

Now, as the `root` user:

```
make DOCUMENTATION_PATH=/usr/share/doc/imagemagick-7.1.1 install
```

To test the installation, issue: `make check`. Note that the EPS, PS, and PDF tests require a working Ghostscript. One test in 'make check' needs "Helvetica" from "Ghostscript Standard Fonts," which are optionally installed in [ghostscript-10.03.1](#) - that test, and one other, might fail, but all the validation can still pass.

Command Explanations

`--enable-hdri`: Enables building of a high dynamic range version of ImageMagick.

`--with-modules`: Enables support for dynamically loadable modules.

`--with-perl`: Enables building and installing of PerlMagick.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-fftw`: Enables support for the FFTW shared library.

`--with-rsvg`: Enables support for the librsvg library.

`--with-autotrace`: Enables support for the Autotrace library.

`--with-wmf`: Enables support for the libwmf library.

`--with-gvc`: Enables support for GraphViz.

`--with-security-policy=open|limited|secure|websafe`: If you wish to enforce a policy other than the default 'open' see [www/security-policy.html](#) in the source for details.

--with-windows-font-dir= <Some/Directory>: This option specifies the directory where the Windows CoreFonts are installed.

--with-dejavu-font-dir= <Some/Directory>: This option specifies the directory where the DejaVu fonts are installed.

The options and parameters listed above are the only ones you should have to pass to the `configure` script to activate all the delegate dependencies. All other dependencies will be automatically detected and utilized in the build if they are installed.

Contents

Installed Programs: magick, Magick++-config, MagickCore-config, and MagickWand-config. (animate, compare, composite, conjure, convert, display, identify, import, magick-script, mogrify, montage, and stream are all symbolic links to magick)

Installed Libraries: libMagickCore-7.Q16HDRI.so, libMagickWand-7.Q16HDRI.so and libMagick++-7.Q16HDRI.so

Installed Directories: /etc/ImageMagick-7, /usr/include/ImageMagick-7, /usr/lib/ImageMagick-7.1.1, /usr/lib/perl5/site_perl/5.40/{auto}/Image/Magick, /usr/share/doc/ImageMagick-7.1.1, and /usr/share/ImageMagick-7

Short Descriptions

animate	animates a sequence of images
compare	compares an image to a reconstructed image
composite	composites various images into the given base image
conjure	processes a MSL script to create an image
convert	converts image(s) from one format to another
display	displays an image
identify	describes the format and characteristics of an image file
import	captures an X window
magick	convert between image formats as well as resize an image, blur, crop, despeckle, dither, draw on, flip, join, re-sample and much more
Magick{++,Core,Wand}-config	show information about the installed versions of the ImageMagick libraries
mogrify	transforms an image
montage	composites various images into a new image
stream	streams one or more pixel components of an image or portion of the image to your choice of storage formats
Image::Magick	allows the reading, manipulation and writing of a large number of image file formats using the ImageMagick library. Run <code>make</code> in the <code>PerlMagick/demo</code> directory of the package source tree after the package is installed to see a nice demo of the module's capabilities
libMagickCore-7.Q16HDRI.so	provides the C API for ImageMagick
libMagickWand-7.Q16HDRI.so	is the recommended C API for ImageMagick. Unlike the MagickCore API it uses only a few opaque types
libMagick++-7.Q16HDRI.so	provides the C++ API for ImageMagick

ISO Codes-4.16.0

Introduction to ISO Codes

The ISO Codes package contains a list of country, language and currency names and it is used as a central database for accessing this data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://ftp.debian.org/debian/pool/main/i/iso-codes/iso-codes_4.16.0.orig.tar.xz
- Download MD5 sum: 5760b5a359301397cc3ab2baa165c31c
- Download size: 3.7 MB
- Estimated disk space required: 95 MB
- Estimated build time: 0.1 SBU (with tests)

Installation of ISO Codes

The directory is called iso-codes-4.16.0.

Install ISO Codes by running the following commands:

```
./configure --prefix=/usr &&  
make
```

To test the results, issue: `make check`.

Note

If you install ISO codes over a previous installed version, the install step will fail when creating some symlinks. In order to properly update them, run:

```
sed -i '/^LN_S/s/s/sfvn/' */Makefile
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: /usr/share/iso-codes, /usr/share/xml/iso-codes

Isof-4.99.0

Introduction to Isof

The Isof package is useful to LiSt Open Files for a given running application or process.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Isof-org/Isof/releases/download/4.99.0/Isof-4.99.0.tar.gz>
- Download MD5 sum: 8c858675f6d6e137df9b4e26ad6c46e8
- Download size: 1.1 MB
- Estimated disk space required: 14 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Isof Dependencies

Required

[libtirpc-1.3.5](#)

Optional

[Nmap-7.95](#) (with a symbolic link `nc` pointing to `ncat` in `/usr/bin`; used in tests)

Kernel Configuration

To run the tests, the following option should be enabled in the kernel configuration:

```
General setup --->  
[*] POSIX Message Queues  
[POSIX_MQUEUE]
```

Installation of Isof

Install lsof by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

The tests should be run as the `root` user. They require that the POSIX message queues are enabled in the kernel, and that [Nmap-7.95](#) be installed with a symbolic link `/usr/bin/nc` pointing to `ncat`.

```
make check
```

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: lsof

Installed Libraries: liblsof.so

Installed Directories: None

Short Descriptions

<code>lsof</code>	lists open files for running processes
<code>liblsof.so</code>	contains an interface for applications to list open files

mandoc-1.14.6

Introduction to mandoc

mandoc is an utility to format manual pages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://mandoc.bsd.lv/snapshots/mandoc-1.14.6.tar.gz>
- Download MD5 sum: f0adf24e8fdef5f3e332191f653e422a
- Download size: 684 KB
- Estimated disk space required: 22 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Installation of mandoc

Many utilities provided by mandoc conflicts with Man-DB in LFS. Only build the `mandoc` command:

```
./configure &&  
make mandoc
```

To test the package, issue: `make regress`.

Now, as the `root` user:

```
install -vm755 mandoc /usr/bin &&  
install -vm644 mandoc.1 /usr/share/man/man1
```

Contents

Installed Programs: mandoc

Short Descriptions

`mandoc` Formats manual pages

pinentry-1.3.1

Introduction to PIN-Entry

The PIN-Entry package contains a collection of simple PIN or pass-phrase entry dialogs which utilize the Assuan protocol as described by the [Ägypten project](#). PIN-Entry programs are usually invoked by the `gpg-agent` daemon, but can be run from the command line as well. There are programs for various text-based and GUI environments, including interfaces designed for Ncurses (text-based), and for the common GTK and Qt toolkits.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.gnupg.org/ftp/gcrypt/pinentry/pinentry-1.3.1.tar.bz2>
- Download MD5 sum: 89a6844fcf76d3c022ce6c6e930c17ee
- Download size: 600 KB
- Estimated disk space required: 17 MB
- Estimated build time: 0.3 SBU

PIN-Entry Dependencies

Required

[libassuan-3.0.1](#) and [libgpg-error-1.50](#)

Optional

[Emacs-29.4](#), [FLTK-1.3.9](#), [Gcr-4.3.0](#) (or [Gcr-3.41.2](#)), [KDE Frameworks-6.5.0](#) (or [kwayland-6.1.4](#) for [lxqt](#)), [libsecret-0.21.4](#), [qt5-components-5.15.14](#), and [efl](#)

Installation of PIN-Entry

Install PIN-Entry by running the following commands:

```
./configure --prefix=/usr      \
            --enable-pinentry-tty \
            --disable-pinentry-qt5 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-pinentry-qt5`: Don't use qt5 even if it is available. Qt5 is deprecated.

`--enable-inside-emacs=yes/no`: Default is no.

`--enable-pinentry-qt=yes/no`: Default is yes.

`--enable-pinentry-gnome3=yes/no`: Default is yes. This option uses [Gcr-4.3.0](#) (or [Gcr-3.41.2](#)) if the former is not installed) for the pinentry dialog.

`--enable-pinentry-tty`: Default is 'maybe'.

Contents

Installed Programs: pinentry (symlink), pinentry-curses, pinentry-emacs, pinentry-fltk, pinentry-gnome3, pinentry-qt, and pinentry-tty

Installed Libraries: None

Installed Directory: None

Short Descriptions

<code>pinentry</code>	is a symbolic link to the default PIN-Entry program
<code>pinentry-curses</code>	is an Ncurses text-based PIN-Entry helper program
<code>pinentry-emacs</code>	is an Emacs version of the PIN-Entry helper program
<code>pinentry-fltk</code>	is a FLTK PIN-Entry helper program
<code>pinentry-gnome3</code>	is a GNOME-3 PIN-Entry helper program
<code>pinentry-qt</code>	is a Qt4 or 5 PIN-Entry helper program
<code>pinentry-tty</code>	is a tty PIN-Entry helper program

Screen-4.9.1

Introduction to Screen

Screen is a terminal multiplexor that runs several separate processes, typically interactive shells, on a single physical character-based terminal. Each virtual terminal emulates a DEC VT100 plus several ANSI X3.64 and ISO 2022 functions and also provides configurable input and output translation, serial port support, configurable logging, multi-user support, and many character encodings, including UTF-8. Screen sessions can be detached and resumed later on a different terminal.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/screen/screen-4.9.1.tar.gz>
- Download MD5 sum: 9a9bdc956bd93e4f0cb9e48678889e26
- Download size: 1020 KB
- Estimated disk space required: 9.5 MB
- Estimated build time: 0.1 SBU

Screen Dependencies

Optional

[Linux-PAM-1.6.1](#)

Installation of Screen

Install Screen by running the following commands:

```
sh autogen.sh &&
./configure --prefix=/usr \
            --infodir=/usr/share/info \
            --mandir=/usr/share/man \
            --with-socket-dir=/run/screen \
            --with-pty-group=5 \
            --with-sys-screenrc=/etc/screenrc &&
sed -i -e "s%usr/local/etc/screenrc%etc/screenrc%" {etc,doc}/* &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -m 644 etc/etcscreenrc /etc/screenrc
```

Command Explanations

`--with-socket-dir=/run/screen`: This option places the per-user sockets in a standard location.

--with-sys-screenrc=/etc/screenrc: This option places the global screenrc file in /etc.

--with-pty-group=5: This option sets the gid to the value used by LFS.

sed -i -e "s%usr/local/etc/screenrc%etc/screenrc%" {etc,doc}/*: This command corrects the configuration and documentation files to the location that is used here for the global screenrc file.

Configuring Screen

Config Files

/etc/screenrc and ~/.screenrc

Configuration Information

You may want to look at the example configuration file that was installed and customize it for your needs.

Contents

Installed Program: screen (symlink) and screen-4.9.1

Installed Libraries: None

Installed Directory: /usr/share/screen and /run/screen

Short Descriptions

screen is a terminal multiplexor with VT100/ANSI terminal emulation

shared-mime-info-2.4

Introduction to Shared Mime Info

The Shared Mime Info package contains a MIME database. This allows central updates of MIME information for all supporting applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/xdg/shared-mime-info/-/archive/2.4/shared-mime-info-2.4.tar.gz>
- Download MD5 sum: aac56db912b7b12a04fb0018e28f2f36
- Download size: 7.5 MB
- Estimated disk space required: 26 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Additional Downloads

- Optional download, required to run the test suite:
<https://anduin.linuxfromscratch.org/BLFS/xdgmime/xdgmime.tar.xz>
xdgmime md5sum: 7dfb4446705d345d3acd672024049e86

Shared Mime Info Dependencies

Required

[GLib-2.80.4](#) and [libxml2-2.13.3](#)

Optional

[xmlto-0.0.29](#)

Installation of Shared Mime Info

Install Shared Mime Info by running the following commands:

If you wish to run the test suite, you must first extract the `xdgmime` tarball into the current directory, and compile it so that `meson` can find it:

```
tar -xf ../xdgmime.tar.xz &&
make -C xdg_mime
```

Now build the package:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D update-mimedb=true .. &&
ninja
```

If you have followed the instructions above to build `xdgmime`, to test the result issue `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D update-mimedb=true`: This parameter tells the build system to run `update-mime-database` during installation. Otherwise, this must be done manually in order to be able to use the MIME database.

Contents

Installed Program: `update-mime-database`

Installed Library: None

Installed Directory: `/usr/share/mime`

Short Descriptions

`update-mime-database` assists in adding MIME data to the database

Sharutils-4.15.2

Introduction to Sharutils

The Sharutils package contains utilities that can create 'shell' archives.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/sharutils/sharutils-4.15.2.tar.xz>
- Download MD5 sum: 5975ce21da36491d7aa6dc2b0d9788e0
- Download size: 1.1 MB
- Estimated disk space required: 22 MB (with the test suite)
- Estimated build time: 0.4 SBU (with the test suite)

Installation of Sharutils

Fix a heap buffer overflow, and an issue exposed by GCC-10:

```
sed -i 's/BUFSIZ/rw_base_size/' src/unshar.c &&
sed -i '/program_name/s/^/extern /' src/*opts.h
```

Install Sharutils by running the following commands:

```
sed -i 's/IO_ftrylockfile/IO_EOF_SEEN/' lib/*.c      &&
echo "#define _IO_IN_BACKUP 0x100" >> lib/stdio-impl.h &&
```

```
./configure --prefix=/usr &&
make
```

To test the results, issue `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: shar, unshar, uudecode and uuencode

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>shar</code>	creates "shell archives" (or shar files) which are in text format and can be mailed
<code>unshar</code>	unpacks a shar file
<code>uudecode</code>	reads a file (or by default the standard input) and writes an encoded version to the standard output. The encoding uses only printing ASCII characters
<code>uuencode</code>	reads a file (or by default the standard input) and decodes the uuencoded version to the standard output

tidy-html5-5.8.0

Introduction to Tidy HTML5

The Tidy HTML5 package contains a command line tool and libraries used to read HTML, XHTML and XML files and write cleaned up markup. It detects and corrects many common coding errors and strives to produce visually equivalent markup that is both W3C compliant and compatible with most browsers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/htacg/tidy-html5/archive/5.8.0/tidy-html5-5.8.0.tar.gz>
- Download MD5 sum: 0f6c55ef651e258adbe5750f555af50f
- Download size: 992 KB
- Estimated disk space required: 11 MB
- Estimated build time: 0.2 SBU

Tidy HTML5 Dependencies

Required

[CMake-3.30.2](#)

Recommended

[libxslt-1.1.42](#)

Installation of Tidy HTML5

Install Tidy HTML5 by running the following commands:

```
cd build/cmake &&
cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TAB2SPACE=ON \
.../.. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
rm -fv /usr/lib/libtidy.a &&
install -v -m755 tab2space /usr/bin
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to build the release library without any debug `assert` in the code.

`-D BUILD_TAB2SPACE=ON`: This switch is used to enable building the `tab2space` utility.

Configuring Tidy HTML5

Config Files

The absolute path of the file specified in `HTML_TIDY`.

Configuration Information

The default configuration options can be set in the file defined in `HTML_TIDY`. Additional configuration options can be passed to `tidy` via command line parameters or the `-config <file>` parameter.

Contents

Installed Programs: `tab2space` and `tidy`

Installed Library: `libtidy.so`

Installed Directory: `/usr/share/doc/tidy-5.8.0`

Short Descriptions

<code>tab2space</code>	is a utility to expand tabs and ensure consistent line endings
<code>tidy</code>	validates, corrects, and pretty-prints HTML files
<code>libtidy.so</code>	library provides the Tidy HTML5 API functions to <code>tidy</code> and can also be called by other programs

Time-1.9

Introduction to Time

The time utility is a program that measures many of the CPU resources, such as time and memory, that other programs use. The GNU version can format the output in arbitrary ways by using a printf-style format string to include various resource measurements.

Although the shell has a builtin command providing similar functionalities, this utility is required by the LSB.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/time/time-1.9.tar.gz>
- Download MD5 sum: d2356e0fe1c0b85285d83c6b2ad51b5f
- Download size: 584 KB
- Estimated disk space required: 4.0 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Installation of Time

Install Time by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: `time`

Installed Library: None

Installed Directory: None

Short Descriptions

`time` reports various statistics about an executed command

tree-2.1.3

Introduction to tree

The tree application is useful to display a dictionary tree's contents, including files, directories, and links.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.com/OldManProgrammer/unix-tree/-/archive/2.1.3/unix-tree-2.1.3.tar.bz2>
- Download MD5 sum: 9be227932ab457c29f33196544cd1e13
- Download size: 56 KB
- Estimated disk space required: 560 KB
- Estimated build time: less than 0.1 SBU

Installation of tree

Install tree by running the following commands:

```
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr MANDIR=/usr/share/man install
```

Contents

Installed Program: `tree`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`tree` displays a directory tree in a terminal

unixODBC-2.3.12

Introduction to unixODBC

The unixODBC package is an Open Source ODBC (Open DataBase Connectivity) sub-system and an ODBC SDK for Linux, Mac OSX, and UNIX. ODBC is an open specification for providing application developers with a predictable API with which to access data sources. Data sources include optional SQL Servers and any data source with an ODBC Driver. unixODBC contains the following components used to assist with the manipulation of ODBC data sources: a driver manager, an

installer library and command line tool, command line tools to help install a driver and work with SQL, drivers and driver setup libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lurcher/unixODBC/releases/download/2.3.12/unixODBC-2.3.12.tar.gz>
- Download MD5 sum: d62167d85bcb459c200c0e4b5a63ee48
- Download size: 1.7 MB
- Estimated disk space required: 28 MB
- Estimated build time: 0.2 SBU (using parallelism=4)

unixODBC Dependencies

Optional

[Mini SQL](#)

Installation of unixODBC

Install unixODBC by running the following commands:

```
./configure --prefix=/usr \
            --sysconfdir=/etc/unixODBC &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&

find doc -name "Makefile*" -delete      &&
chmod 644 doc/{lst,ProgrammerManual/Tutorial}/*  &&

install -v -m755 -d /usr/share/doc/unixODBC-2.3.12 &&
cp      -v -R doc/* /usr/share/doc/unixODBC-2.3.12
```

Command Explanations

`--enable-drivers`: This parameter enables building the drivers that were installed by default in previous versions.

`--enable-drivers-conf`: This parameter enables building the driver configuration libraries that were installed by default in previous versions.

Configuring unixODBC

Config Files

`/etc/unixODBC/*`

Configuration Information

The files in `/etc/unixODBC` are meant to be configured by the system administrator (or the ODBC site administrator if appropriate privileges are granted to `/etc/unixODBC`). These files are not meant to be directly edited. The ODBC installer library is responsible for reading and writing the unixODBC config files.

Unfortunately, there are not many `man`, or any `info` pages for the various programs available in the unixODBC package. Along with the information in the "Short Descriptions" below and the documentation installed in `/usr/share/doc/unixODBC-2.3.12`, there are many `README` files throughout the source tree where the use and functionality of the programs can be found. Additionally, you can use the parameter `-?` for syntax and usage information. Lastly, the unixODBC web site at <http://www.unixodbc.org/> has very good information.

Contents

Installed Programs: `dltest`, `isql`, `iusql`, `odbc_config`, `odbcinst`, and `slencheck`

Installed Libraries: libodbc.so, libodbccr.so, and libodbcinst.so
Installed Directories: /etc/unixODBC and /usr/share/doc/unixODBC-2.3.12

Short Descriptions

<code>dltest</code>	is a utility used to check a shared library to see if it can be loaded and if a given symbol exists in it
<code>isql</code>	is a utility which can be used to submit SQL to a data source and to format/output results. It can be used in batch or interactive mode
<code>isql</code>	provides the same functionality as the <code>isql</code> program
<code>odbc_config</code>	is used to find out details about the installation of the unixODBC package
<code>odbcinst</code>	is a utility created for install script/RPM writers. It is a command line interface to key functionality in the <code>libodbcinst</code> library. It does not copy any files (i.e., libraries) but it will modify the ODBC System Information for the user
<code>slencheck</code>	is a utility which attempts to check whether an ODBC driver was built with 32-bit or 64-bit SQLLEN types

xdg-dbus-proxy-0.1.5

Introduction to xdg-dbus-proxy

The `xdg-dbus-proxy` package contains a filtering proxy for D-Bus connections. This is useful for forwarding data in and out of a sandbox.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/flatpak/xdg-dbus-proxy/releases/download/0.1.5/xdg-dbus-proxy-0.1.5.tar.xz>
- Download (FTP):
- Download MD5 sum: b496f34b0fa4747a66eb5adb63b7d626
- Download size: 124 KB
- Estimated disk space required: 1.4 MB
- Estimated build time: less than 0.1 SBU

xdg-dbus-proxy Dependencies

Required

[GLib-2.80.4](#)

Installation of xdg-dbus-proxy

Install `xdg-dbus-proxy` by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: xdg-dbus-proxy

Installed Libraries: None

Installed Directories: None

Short Descriptions

`xdg-dbus-proxy` is a filtering proxy for D-Bus connections

Xdg-user-dirs-0.18

Introduction to Xdg-user-dirs

Xdg-user-dirs is a tool to help manage “well known” user directories like the desktop folder and the music folder. It also handles localization (i.e. translation) of the filenames.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://user-dirs.freedesktop.org/releases/xdg-user-dirs-0.18.tar.gz>
- Download MD5 sum: dc7decea7ffb58cd067eff1fe1798cae
- Download size: 267 KB
- Estimated disk space required: 3.0 MB
- Estimated build time: less than 0.1 SBU

Xdg-user-dirs Dependencies

Optional

`docbook-xml-4.5`, `docbook-xsl-nons-1.79.2`, and `libxslt-1.1.42` (all three for building the man pages)

Installation of Xdg-user-dirs

Install xdg-user-dirs by running the following commands:

```
./configure --prefix=/usr \
            \
            \
            \
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-documentation`: Disable the installation of the man pages. Remove this switch if you've installed the optional dependencies and you wish to install the man pages of this package.

Configuring Xdg-user-dirs

Config Files

`~/.config/user-dirs.dirs`, `/etc/xdg/user-dirs.conf`, and `/etc/xdg/user-dirs.defaults`. Those default locations can be overridden by `XDG_CONFIG_HOME` and `XDG_CONFIG_DIRS`

Contents

Installed Programs: xdg-user-dir and xdg-user-dirs-update

Installed Library: None

Installed Directory: None

Short Descriptions

<code>xdg-user-dir</code>	looks up the current path for one of the XDG user directories
<code>xdg-user-dirs-update</code>	creates localized versions of the user directories

Chapter 12. System Utilities

This chapter contains mainly hardware utilities. It also contains some applications used by other applications in the book for installation or configuration purposes.

AccountsService-23.13.9

Introduction to AccountsService

The AccountsService package provides a set of D-Bus interfaces for querying and manipulating user account information and an implementation of those interfaces based on the [usermod\(8\)](#), [useradd\(8\)](#), and [userdel\(8\)](#) commands.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/accountsservice/accountsservice-23.13.9.tar.xz>
- Download MD5 sum: 03dccfe1b306b7ca19743e86d118e64d
- Download size: 621 KB
- Estimated disk space required: 9.8 MB (add 0.5 MB for tests)
- Estimated build time: less than 0.1 SBU (with tests)

AccountsService Dependencies

Required

[Polkit-125](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [Systemd-256.4](#) (runtime), and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#) and [xmlto-0.0.29](#)

Optional (for tests)

[dbusmock-0.32.1](#) and [PyGObject-3.48.2](#)

Installation of AccountsService

First, rename a directory whose presence prevents the build system from running if [dbusmock-0.32.1](#) is not installed:

```
mv tests/dbusmock{,-tests}
```

Then fix a test script so that the new directory is found, and adapt it for Python 3.12.0 or later:

```
sed -e '/accounts_service\.py/s/dbusmock/dbusmock-tests/' \
-e 's/assertEquals/assertEqual/' \
-i tests/test-libaccounts.service.py
```

Fix one test that fails if the `en_IE.UTF-8` locale is not installed:

```
sed -i '/^SIMULATED_SYSTEM_LOCALE/s/en_IE.UTF-8/en_HK.iso88591/' tests/test-daemon.py
```

Install AccountsService by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D admin_group=adm
```

Now adapt the shipped mocklibc copy to allow building the test suite with GCC 14 or later:

```
grep 'print_indent' ..subprojects/mocklibc-1.0/src/netgroup.c \
    | sed 's/ {/;/' >> ..subprojects/mocklibc-1.0/src/netgroup.h &&
sed -i 'li#include <stdio.h>' \
    ..subprojects/mocklibc-1.0/src/netgroup.h
```

Build the package:

```
ninja
```

To test the package, issue `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D admin_group=adm`: This switch sets the group for administrator accounts.

`-D docbook=true`: This switch enables building the D-Bus interface API documentation (needs [xmlto-0.0.29](#)).

`-D gtk_doc=true`: This switch enables building the libaccounts-service API documentation (needs [GTK-Doc-1.34.0](#)).

`-D vapi=false`: This switch disables building the vala bindings. Use it if you have not installed [Vala-0.56.17](#).

Configuring AccountsService

To allow users in the adm group to be listed as Administrators, execute the following commands as the `root` user:

```
cat > /etc/polkit-1/rules.d/40-adm.rules << "EOF"
polkit.addAdminRule(function(action, subject) {
    return ["unix-group:adm"];
});
EOF
```

Contents

Installed Programs: accounts-daemon (in `/usr/libexec`)

Installed Libraries: libaccounts-service.so

Installed Directories: /usr/include/accounts-service-1.0, /usr/share/accounts-service, /usr/share/gtk-doc/html/libaccounts-service (optional), and /var/lib/AccountsService

Short Descriptions

<code>accounts-daemon</code>	is the AccountsService daemon
<code>libaccounts-service.so</code>	contains the AccountsService API functions

acpid-2.0.34

Introduction to acpid

The acpid (Advanced Configuration and Power Interface event daemon) is a completely flexible, totally extensible daemon for delivering ACPI events. It listens on netlink interface and when an event occurs, executes programs to handle the event.

The programs it executes are configured through a set of configuration files, which can be dropped into place by packages or by the user.

Note

Some other packages may handle some ACPI events as well and they may conflict with this package. For example, [Systemd-256.4](#) (read the documentation for `Handle*=` in [logind.conf\(5\)](#) for details) and [UPower-1.90.4](#) (used by many desktop environments such as GNOME, KDE, and XFCE for handling ACPI events). If you've installed such a package and it's enough for your use case, this package is probably not needed. If you really need this package, you must be careful configuring it and the other packages handling ACPI events to avoid conflicts. Notably, [Systemd-256.4](#) handles some ACPI events by default, so the handling of these events by [Systemd-256.4](#) should be disabled first if handling these events with acpid (again, read [logind.conf\(5\)](#) for details).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/acpid2/acpid-2.0.34.tar.xz>
- Download MD5 sum: 988c2e3fd5ba0ea4492d3ba7273af295
- Download size: 160 KB
- Estimated disk space required: 1.9 MB
- Estimated build time: less than 0.1 SBU

Installation of acpid

Install acpid by running the following commands:

```
./configure --prefix=/usr \
            --docdir=/usr/share/doc/acpid-2.0.34 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
install -v -m755 -d /etc/acpi/events &&
cp -r samples /usr/share/doc/acpid-2.0.34
```

Configuring acpid

acpid is configured by user defined events. Place event files under `/etc/acpi/events` directory. If an event occurs, acpid recurses through the event files in order to see if the regex defined after "event" matches. If they do, action is executed.

The following brief example will suspend the system when the laptop lid is closed. The example also disables the default handling of the lid close event by [Systemd-256.4](#) when the system is on battery and not connected to any external monitor, in order to avoid a conflict:

```
cat > /etc/acpi/events/lid << "EOF"
event=button/lid
action=/etc/acpi/lid.sh
EOF

cat > /etc/acpi/lid.sh << "EOF"
#!/bin/sh
/bin/grep -q open /proc/acpi/button/lid/LID/state && exit 0
/usr/bin/systemctl suspend
EOF
chmod +x /etc/acpi/lid.sh

mkdir -pv /etc/systemd/logind.conf.d
echo HandleLidSwitch=ignore > /etc/systemd/logind.conf.d/acpi.conf
```

Unfortunately, not every computer labels ACPI events in the same way (for example, the lid may be recognized as `LID0` instead of `LID`). To determine how your buttons are recognized, use the `acpi_listen` tool. Also, look in the `samples` directory under `/usr/share/doc/acpid-2.0.34` for more examples.

Systemd Socket

To start the `acpid` daemon at boot, install the systemd unit from the [blfs-systemd-units-20240801](#) package by running the following command as the `root` user:

```
make install-acpid
```

Note

This package uses socket based activation and will be started when something needs it. No standalone unit file is provided for this package.

Contents

Installed Programs: `acpid`, `acpi_listen`, and `kacpimon`

Installed Libraries: None

Installed Directories: /etc/acpi and /usr/share/doc/acpid-2.0.34

Short Descriptions

<code>acpid</code>	is a program that listens for ACPI events and executes the rules that match the received event
<code>acpi_listen</code>	is a simple tool which connects to <code>acpid</code> and listens for events
<code>kacpimon</code>	is a monitor program that connects to three sources of ACPI events (events file, netlink and input layer) and then reports on what it sees while it is connected

at-3.2.5

Introduction to at

The `at` package provides delayed job execution and batch processing. It is required for Linux Standards Base (LSB) conformance.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://anduin.linuxfromscratch.org/BLFS/at/at_3.2.5.orig.tar.gz
- Download MD5 sum: ca3657a1c90d7c3d252e0bc17feddc6e
- Download size: 130 KB
- Estimated disk space required: 1.9 MB (including tests)
- Estimated build time: less than 0.1 SBU

at Dependencies

Required

An [MTA](#)

Optional

[Linux-PAM-1.6.1](#)

Installation of at

Before building `at`, as the `root` user you should create the group and user `atd` which will run the `atd` daemon:

```
groupadd -g 17 atd
useradd -d /dev/null -c "atd daemon" -g atd -s /bin/false -u 17 atd
```

Install `at` with the following commands:

```

./configure --with-daemon_username=atd      \
--with-daemon_groupname=atd                \
SENDMAIL=/usr/sbin/sendmail               \
--with-jobdir=/var/spool/atjobs           \
--with-atspool=/var/spool/atspool          \
--with-systemdsystemunitdir=/lib/systemd/system &&
make -j1

```

To test the results, issue: `make test`.

Now, as the `root` user:

```

make install docdir=/usr/share/doc/at-3.2.5 \
atdocdir=/usr/share/doc/at-3.2.5

```

Configuring at

Config Files

`/etc/at.allow` and `/etc/at.deny` determines who can submit jobs via at or batch.

Linux PAM Configuration

If At has been built with Linux PAM support, you need to create a PAM configuration file, to get it working correctly with BLFS.

Issue the following commands as the `root` user to create the configuration file for Linux PAM:

```

cat > /etc/pam.d/atd << "EOF"
# Begin /etc/pam.d/atd

auth    required pam_unix.so
account required pam_unix.so
password required pam_unix.so
session required pam_unix.so

# End /etc/pam.d/atd
EOF

```

Systemd Unit

To start the `atd` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable atd
```

Contents

Installed Programs: at, atd, atq (symlink), atrm (symlink), atrun, and batch

Installed Libraries: None

Installed Directories: /usr/share/doc/at-3.2.5

Short Descriptions

<code>at</code>	queues, examines or deletes jobs for later execution
<code>atd</code>	is the daemon that runs jobs queued for later execution
<code>atq</code>	lists the user's pending jobs, or all jobs, if superuser
<code>atrm</code>	deletes jobs, identified by their job number
<code>atrun</code>	runs jobs queued for later execution
<code>batch</code>	is a script that executes commands when system load levels permit

autofs-5.1.9

Introduction to Autofs

Autofs controls the operation of the automount daemons. The automount daemons automatically mount filesystems when they are accessed and unmount them after a period of inactivity. This is done based on a set of pre-configured maps.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/daemons/autofs/v5/autofs-5.1.9.tar.xz>
- Download MD5 sum: 06fb59a03c82364a0d788435b6853d70
- Download size: 328 KB
- Estimated disk space required: 5.7 MB
- Estimated build time: less than 0.1 SBU

Autofs Dependencies

Required

[libtirpc-1.3.5](#) and [rpcsvc-proto-1.4.4](#)

Optional

[nfs-utils-2.6.4](#), [libnsl-2.0.1](#), [libxml2-2.13.3](#), [MIT Kerberos V5-1.21.3](#), [OpenLDAP-2.6.8](#) (client only), and [Cyrus SASL-2.1.28](#)

Kernel Configuration

Verify that automounter kernel support has been enabled:

```
File systems --->
  <*/M> Kernel automounter support (supports v3, v4 and v5)           [AUTOFS_FS]
```

Optionally, enable the following options in the kernel configuration:

```
File systems --->
  [*] Network File Systems --->                                [NETWORK_FILESYSTEMS]
    <*/M> NFS client support                                     [NFS_FS]
    <*/M> SMB3 and CIFS support (advanced network filesystem)   [CIFS]
```

Recompile and install the new kernel, if necessary.

Installation of Autofs

Install Autofs by running the following commands:

```
./configure --prefix=/usr
            \
            --with-mapdir=/etc/autofs \
            \
            --with-libtirpc \
            \
            --with-systemd \
            \
            --without-openldap \
            \
            --mandir=/usr/share/man &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Caution

If autofs is already installed on your system, be sure to backup your configuration files. They'll be overwritten by the following command.

Install the default configuration files, still as the `root` user:

```
make install_samples
```

Command Explanations

--with-libtirpc: This switch forces the package to use libtirpc for RPC functionality instead of relying on implementation from Glibc, which was removed in LFS 8.1.

--with-systemd: This switch enables installation of the bundled systemd units.

--without-openldap: This switch disables openldap if found. If openldap is desired, omit this switch. Note that openldap support in autofs requires [MIT Kerberos V5-1.21.3](#).

Configuring Autofs

Config Files

/etc/sysconfig/autofs.conf, /etc/autofs/auto.master, /etc/autofs/auto.misc, and /etc/autofs/auto.net

Configuration Information

The installation process creates auto.master, auto.misc, auto.smb, and auto.net. Replace the auto.master file with the following commands as the `root` user:

```
mv /etc/autofs/auto.master /etc/autofs/auto.master.bak &&
cat > /etc/autofs/auto.master << "EOF"
# Begin /etc/autofs/auto.master

/media/auto  /etc/autofs/auto.misc --ghost
#/home       /etc/autofs/auto.home

# End /etc/autofs/auto.master
EOF
```

This file creates a new media directory, `/media/auto` that will overlay any existing directory of the same name. In this example, the file, `/etc/autofs/auto.misc`, has a line:

```
cd -fstype=iso9660,ro,nosuid,nodev :/dev/cdrom
```

that will mount a cdrom as `/media/auto/cd` if that directory is accessed. The `--ghost` option tells the automounter to create "ghost" versions (i.e. empty directories) of all the mount points listed in the configuration file regardless whether any of the file systems are actually mounted or not. This is very convenient and highly recommended, because it will show you the available auto-mountable file systems as existing directories, even when their file systems aren't currently mounted. Without the `--ghost` option, you'll have to remember the names of the directories. As soon as you try to access one of them, the directory will be created and the file system will be mounted. When the file system gets unmounted again, the directory is destroyed too, unless the `--ghost` option was given.

Note

An alternative method would be to specify another automount location such as `/var/lib/auto/cdrom` and create a symbolic link from `/media/cdrom` to the automount location.

The `auto.misc` file must be configured to your working hardware. The loaded configuration file should load your cdrom if `/dev/cdrom` is active or it can be edited to match your device setup. Examples for floppies are available in the file and easily activated. Documentation for this file is available using the `man 5 autofs` command.

In the second line, if enabled, a user's home directory would be mounted via NFS upon login. The `/etc/home.auto` would need to exist and have an entry similar to:

```
joe example.org:/export/home/joe
```

where the directory `/export/home/joe` is exported via NFS from the system `example.org`. NFS shares are covered on the next page.

This package could also be used to mount SMB shares, however that feature is not configured in these instructions. For additional configuration information, see the man pages for [`auto.master\(5\)`](#). There are also web resources such as this [AUTofs HOWTO](#) available.

Systemd Unit

To start Autofs at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable autofs
```

Note

You can also specify `OPTIONS` variable in the `/etc/sysconfig/autofs` file with any additional parameters that you might want to pass to the automount daemon.

Contents

Installed Program: `automount`

Installed Libraries: `libautofs.so`, `lookup_dir.so`, `lookup_file.so`, `lookup_files.so` (symlink to `lookup_file.so`), `lookup_hosts.so`, `lookup_ldap.so` (optional), `lookup_multi.so`, `lookup_nis.so` (symlink to `lookup_yp.so`), `lookup_program.so`, `lookup_userhome.so`, `lookup_yp.so`, `mount_afs.so`, `mount_automount.so`, `mount_bind.so`, `mount_changer.so`, `mount_ext2.so`, `mount_ext3.so` (symlink to `mount_ext2.so`), `mount_ext4.so` (symlink to `mount_ext2.so`), `mount_generic.so`, `mount_nfs.so`, `mount_nfs4.so` (symlink to `mount_nfs.so`), `parse_amd.so`, and `parse_sun.so`

Installed Directories: `/usr/lib/autofs` and `/etc/autofs`

Short Descriptions

`automount` is the daemon that performs the mounting when a request is made for the device

BlueZ-5.77

Introduction to BlueZ

The BlueZ package contains the Bluetooth protocol stack for Linux.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/bluetooth/bluez-5.77.tar.xz>
- Download MD5 sum: 02878560e89165540c4a0b8c060d1092
- Download size: 2.4 MB
- Estimated disk space required: 168 MB (with tests)
- Estimated build time: 0.7 SBU (using parallelism=4; with tests)

BlueZ Dependencies

Required

[dbus-1.14.10](#), [GLib-2.80.4](#), and [libical-3.0.18](#)

Optional

[docutils-0.21.2](#) (to generate man pages)

Kernel Configuration

If you are building this package to use bluetooth devices (rather than as a build dependency), enable the following options in the kernel configuration, also the options in the "Cryptographic API" section if you intend to run the tests, and recompile the kernel if necessary:

```
General setup --->
  # If it is disabled, [TIMERFD] and [EVENTFD] will be hidden and
  # enabled implicitly. We DO NOT recommend to enable [EXPERT]
  # unless you are really an expert user:
  [*] Configure standard kernel features (expert users) --->      [EXPERT]
    [*] Enable timerfd() system call                                [TIMERFD]
    [*] Enable eventfd() system call                                 [EVENTFD]

  [*] Networking support --->                                     [NET]
    <*/M> Bluetooth subsystem support --->                         [BT]
```

```

[*] Bluetooth Classic (BR/EDR) features [BT_BREDR]
<*/M> RFCOMM protocol support [BT_RFCOMM]
[*] RFCOMM TTY support [BT_RFCOMM_TTY]
<*/M> BNEP protocol support [BT_BNEP]
[*] Multicast filter support [BT_BNEP_MC_FILTER]
[*] Protocol filter support [BT_BNEP_PROTO_FILTER]
<*/M> HIDP protocol support [BT_HIDP]
Bluetooth device drivers --->
# Select the appropriate drivers for your bluetooth hardware.
# There are more vendor-specific drivers not listed here:
< /*/M> HCI USB driver [BT_HCIBTUSB]
< /*/M> HCI SDIO driver [BT_HCIBTSDIO]
< /*/M> HCI UART driver [BT_HCIUART]
[RFKILL]

<*/M> RF switch subsystem support ---> [RFKILL]

--> Cryptographic API ---> [CRYPTO]
Crypto core or helper --->
<*/M> Userspace cryptographic algorithm configuration [CRYPTO_USER]
Block ciphers --->
<*/M> AES (Advanced Encryption Standard) [CRYPTO_AES]
AEAD (authenticated encryption with associated data) ciphers --->
<*/M> CCM (Counter with Cipher Block Chaining-MAC) [CRYPTO_CCM]
Hashes, digests, and MACs --->
<*/M> CMAC (Cipher-based MAC) [CRYPTO_CMAC]
Userspace interface --->
<*/M> Hash algorithms [CRYPTO_USER_API_HASH]
<*/M> Symmetric key cipher algorithms [CRYPTO_USER_API_SKCIPHER]
<*/M> AEAD cipher algorithms [CRYPTO_USER_API_AEAD]

```

Installation of BlueZ

Install BlueZ by running the following commands:

```

./configure --prefix=/usr \
--sysconfdir=/etc \
--localstatedir=/var \
--disable-manpages \
--enable-library &&
make

```

To test the results, issue: `make check`.

Now, as the `root` user:

```

make install &&
ln -svf ../libexec/bluetooth/bluetoothd /usr/sbin

```

Install the main configuration file as the `root` user:

```

install -v -dm755 /etc/bluetooth &&
install -v -m644 src/main.conf /etc/bluetooth/main.conf

```

If desired, install the API documentation as the `root` user:

```

install -v -dm755 /usr/share/doc/bluez-5.77 &&
install -v -m644 doc/*.txt /usr/share/doc/bluez-5.77

```

Command Explanations

`--disable-manpages`: This switch disables generating the manual pages because of the reliance on 'rst2man' in docutils. Remove this switch if you have [docutils-0.21.2](#) installed and wish to generate the manual pages.

`--enable-library`: This switch enables building the BlueZ 4 compatibility library which is required by some applications.

`ln -svf ../libexec/bluetooth/bluetoothd /usr/sbin`: This command makes access to the bluetooth daemon more convenient.

Configuring BlueZ

Configuration Files

`/etc/bluetooth/main.conf` is installed automatically during the installation. Additionally, there are two supplementary configuration files. You can optionally install the following files as the `root` user:

```
cat > /etc/bluetooth/rfcomm.conf << "EOF"
# Start rfcomm.conf
# Set up the RFCOMM configuration of the Bluetooth subsystem in the Linux kernel.
# Use one line per command
# See the rfcomm man page for options

# End of rfcomm.conf
EOF
```

```
cat > /etc/bluetooth/uart.conf << "EOF"
# Start uart.conf
# Attach serial devices via UART HCI to BlueZ stack
# Use one line per device
# See the hcitattach man page for options

# End of uart.conf
EOF
```

Systemd Bluez Services

To start the `bluetoothd` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable bluetooth
```

To start the `obexd` daemon for a user session (to support some Bluetooth programs using it), enable the previously installed systemd unit for all users by running the following command as the `root` user:

```
systemctl enable --global obex
```

Note

Systemd will start the Bluetooth daemon only when a bluetooth device is detected on the system.

Contents

Installed Programs: bluemoon, bluetoothctl, bluetoothd (symlink), btattach, btmon, hex2hcd, l2ping, l2test, mpris-proxy, and rctest

Installed Library: libbluetooth.so

Installed Directories: /etc/bluetooth, /usr/{include,libexec}/bluetooth, and /usr/share/doc/bluez-5.77

Short Descriptions

<code>bluemoon</code>	is a Bluetooth configuration utility
<code>bluetoothctl</code>	is the interactive Bluetooth control program
<code>bluetoothd</code>	is the Bluetooth daemon
<code>btmon</code>	provides access to the Bluetooth subsystem monitor infrastructure for reading HCI traces
<code>hex2hcd</code>	is used to convert a file needed by Broadcom devices to hcd (Broadcom bluetooth firmware) format
<code>l2ping</code>	is used to send a L2CAP echo request to the Bluetooth MAC address given in dotted hex notation
<code>l2test</code>	is a L2CAP testing program
<code>rctest</code>	is used to test RFCOMM communications on the Bluetooth stack
<code>libbluetooth.so</code>	contains the BlueZ 4 API functions

Bubblewrap-0.9.0

Introduction to Bubblewrap

Bubblewrap is a setuid implementation of user namespaces, or sandboxing, that provides access to a subset of kernel user namespace features. Bubblewrap allows user owned processes to run in an isolated environment with limited access to the underlying filesystem.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/containers/bubblewrap/releases/download/v0.9.0/bubblewrap-0.9.0.tar.xz>
- Download MD5 sum: adcbd7c08ac068a9328ec93cd83716e5
- Download size: 120 KB
- Estimated disk space required: 3.3 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Bubblewrap Dependencies

Optional

[libxslt-1.1.42](#) (to generate manual pages) and [libseccomp-2.5.5](#) (built with python bindings, for tests)

Kernel Configuration

When this package began, upstream expected it could be installed suid-root. That was a long time ago, suid-root is generally considered a bad idea. As well as the default namespaces, this package requires the optional User namespace to be enabled. If that has not yet been enabled, select the following option in the kernel configuration and recompile the kernel:

```
General setup --->
  *-- Namespaces support --->
    [*] User namespace                                [NAMESPACES]
                                                [USER_NS]
```

Installation of Bubblewrap

Install Bubblewrap by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

Next, if you desire to run the test suite, fix an issue caused by the merged-/usr configuration in LFS:

```
sed 's@symlink usr/lib64@ro-bind-try /lib64@' -i ..../tests/libtest.sh
```

To test the results, issue (as a user other than the `root` user): `ninja test`

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: bwrap

Installed Libraries: None

Installed Directories: None

Short Descriptions

`bwrap` generates a sandbox for a program to run in

Colord-1.4.7

Introduction to Colord

Colord is a system service that makes it easy to manage, install, and generate color profiles. It is used mainly by GNOME Color Manager for system integration and use when no users are logged in.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/colord/releases/colord-1.4.7.tar.xz>
- Download MD5 sum: 94bd795efa1931a34990345e4ac439a8
- Download size: 1.7 MB
- Estimated disk space required: 18 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/colord-1.4.7-upstream_fixes-1.patch

Colord Dependencies

Required

[dbus-1.14.10](#), [GLib-2.80.4](#) (GObject Introspection recommended), [Little CMS-2.16](#), [Polkit-125](#), and [SQLite-3.46.1](#)

Recommended

[libgudev-238](#), [libgusb-0.4.9](#), [Systemd-256.4](#) (runtime), and [Vala-0.56.17](#)

Optional

[gnome-desktop-44.1](#) and [colord-gtk-0.3.1](#) (to build the example tools), [DocBook-utils-0.6.14](#), [GTK-Doc-1.34.0](#), [libxslt-1.1.42](#), [SANE-1.2.1](#), [ArgyllCMS](#), and [Bash Completion](#)

Installation of Colord

First, fix a misuse of the sqlite API causing a test failure and potential runtime crash, as well as an overly-strict systemd hardening setting causing the `systemd` service to fail to start:

```
patch -Np1 -i ../../colord-1.4.7-upstream_fixes-1.patch
```

There should be a dedicated user and group to take control of the colord daemon after it has started. Issue the following commands as the `root` user:

```
groupadd -g 71 colord &&
useradd -c "Color Daemon Owner" -d /var/lib/colord -u 71 \
-g colord -s /bin/false colord
```

Install Colord by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
--prefix=/usr \
--buildtype=release \
-D daemon_user=colord \
-D vapi=true \
-D systemd=true \
-D libcolordcompat=true \
-D argyllcms_sensor=false \
-D bash_completion=false \
-D docs=false \
-D man=false \
&&
ninja
```

Now, as the `root` user:

```
ninja install
```

To test the results, issue: `ninja test`. One test, `colord-self-test-daemon`, will fail if the package is already installed. The test suite must be run with the system-wide D-Bus Daemon running.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D daemon_user=colord`: This switch is used so the colord daemon will run as an unprivileged user instead of the `root` user.

`-D vapi=true`: This switch enables building the Vala bindings. Remove if you don't have [Vala-0.56.17](#) installed.

`-D systemd=true`: This switch enables installation of the `systemd` service.

`-D libcolordcompat=true`: This switch enables building a compatibility library for older packages that use Colord.

`-D argyllcms_sensor=false`: This switch disables the ArgLLCMS sensor driver. Omit if you have [ArgylicMS](#) installed and wish to use it.

`-D bash_completion=false`: This switch disables Bash Completion support for Colord applications.

`-D docs=false`: This switch disables building of documentation. Omit if you have [GTK-Doc-1.34.0](#) available.

`-D man=false`: This switch disables building of man pages. Omit if you have [DocBook-utils-0.6.14](#) available. Namespaced versions of the docbook-xsl stylesheets are also required.

Contents

Installed Programs: cd-create-profile, cd-fix-profile, cd-iccdump, cd-it8, and colormgr

Installed Libraries: libcolord.so, libcolordcompat.so, libcolordprivate.so, and libcolorhug.so

Installed Directories: /usr/include/colord-1, /usr/lib/colord-{plugins,sensors}, /usr/share/color{d}, /usr/share/gtk-doc/html/colord, and /var/lib/colord

Short Descriptions

<code>cd-create-profile</code>	is the Color Manager Profile Creation Tool
<code>cd-fix-profile</code>	is a tool used to fix metadata in ICC profiles
<code>cd-iccdump</code>	dumps the contents of an ICC profile as human readable text
<code>cd-it8</code>	is the Color Manager Testing Tool
<code>colormgr</code>	is a text-mode program that allows you to interact with colord on the command line
<code>libcolord.so</code>	contains the Colord API functions
<code>libcolordcompat.so</code>	contains legacy API functions for compatibility with older applications
<code>libcolordprivate.so</code>	contains internal API functions for the programs included with Colord
<code>libcolorhug.so</code>	contains a simple display hardware colorimeter

cpio-2.15

Introduction to cpio

The cpio package contains tools for archiving.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/cpio/cpio-2.15.tar.bz2>
- Download MD5 sum: 3394d444ca1905ea56c94b628b706a0b
- Download size: 1.6 MB
- Estimated disk space required: 21 MB (with tests and docs)
- Estimated build time: 0.3 SBU (with tests and docs)

CPIO Dependencies

Optional

[texlive-20240312](#) (or [install-tl-unx](#))

Installation of cpio

Install cpio by running the following commands:

```
./configure --prefix=/usr \
    --enable-mt \
    --with-rmt=/usr/libexec/rmt &&
make &&
makeinfo --html          -o doc/html      doc/cpio.texi &&
makeinfo --html --no-split -o doc/cpio.html doc/cpio.texi &&
makeinfo --plaintext      -o doc/cpio.txt   doc/cpio.texi
```

If you have [texlive-20240312](#) installed and wish to create PDF or Postscript documentation, issue one or both of the following commands:

```
make -C doc pdf &&
make -C doc ps
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
install -v -m755 -d /usr/share/doc/cpio-2.15/html &&
install -v -m644   doc/html/* \
                  /usr/share/doc/cpio-2.15/html &&
install -v -m644   doc/cpio.{html,txt} \
                  /usr/share/doc/cpio-2.15
```

If you built PDF or Postscript documentation, install it by issuing the following commands as the `root` user:

```
install -v -m644 doc/cpio.{pdf,ps,dvi} \
                 /usr/share/doc/cpio-2.15
```

Command Explanations

`--enable-mt`: This parameter forces the building and installation of the `mt` program.

`--with-rmt=/usr/libexec/rmt`: This parameter inhibits building the `rmt` program as it is already installed by the Tar package in LFS.

Contents

Installed Programs: cpio and mt

Installed Libraries: None

Installed Directories: /usr/share/doc/cpio-2.15

Short Descriptions

<code>cpio</code>	copies files to and from archives
<code>mt</code>	controls magnetic tape drive operations

cups-pk-helper-0.2.7

Introduction to cups-pk-helper

The cups-pk-helper package contains a PolicyKit helper used to configure Cups with fine-grained privileges.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/cups-pk-helper/releases/cups-pk-helper-0.2.7.tar.xz>
- Download MD5 sum: 0cdadec9ea8f88b7fc7af8ca206da2bd
- Download size: 56 KB
- Estimated disk space required: 5.3 MB
- Estimated build time: 0.1 SBU

cups-pk-helper Dependencies

Required

[Cups-2.4.10](#) and [Polkit-125](#)

Installation of cups-pk-helper

Install cups-pk-helper by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: cups-pk-helper-mechanism (library executable)

Installed Libraries: None

Installed Directories: None

dbus-1.14.10

Introduction to D-Bus

Even though D-Bus was built in LFS, there are some features provided by the package that other BLFS packages need, but their dependencies didn't fit into LFS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://dbus.freedesktop.org/releases/dbus/dbus-1.14.10.tar.xz>
- Download MD5 sum: 46070a3487817ff690981f8cd2ba9376
- Download size: 1.3 MB
- Estimated disk space required: 23 MB (add 25 MB for tests)
- Estimated build time: 0.2 SBU (add 1.5 SBU for tests)

D-Bus Dependencies

Recommended

[Xorg Libraries](#) (for `dbus-launch` program)

Optional

[Systemd-256.4](#) (runtime, for registering services launched by D-Bus session daemon as systemd user services); For the tests: [D-Bus Python-1.3.2](#), [PyGObject-3.48.2](#), and [Valgrind-3.23.0](#); for documentation: [Doxygen-1.12.0](#), [xmlto-0.0.29](#), [Ducktype](#), and [Yelp Tools](#)

Installation of D-Bus

Install D-Bus by running the following commands (you may wish to review the output from `./configure --help` first and add any desired parameters to the `configure` command shown below):

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --localstatedir=/var \
            --runstatedir=/run \
            --enable-user-session \
            --disable-doxygen-docs \
            --disable-xml-docs \
            --disable-static \
            --docdir=/usr/share/doc/dbus-1.14.10 \
            --with-system-socket=/run/dbus/system_bus_socket &&
make
```

See below for test instructions.

Now, as the `root` user:

```
make install
```

If you are using a DESTDIR install, `dbus-daemon-launch-helper` needs to be fixed afterwards. Issue, as `root` user:

```
chown -v root:messagebus /usr/libexec/dbus-daemon-launch-helper &&
chmod -v      4750          /usr/libexec/dbus-daemon-launch-helper
```

Many tests are disabled unless both [D-Bus Python-1.3.2](#) and [PyGObject-3.48.2](#) have been installed. They must be run as an unprivileged user from a local session with bus address. To run the standard tests issue `make check`.

If you want to run the unit regression tests, `configure` requires additional parameters which expose additional functionality in the binaries that are not intended to be used in a production build of D-Bus. If you would like to run the tests, issue the following commands (for the tests, you don't need to build the docs):

```
make distclean &&
PYTHON=python3 ./configure --enable-tests \
                        --enable-asserts \
                        --disable-doxygen-docs \
                        --disable-xml-docs &&
make &&
make check
```

One test, `test-autolaunch`, is known to fail. There have also been reports that the tests may fail if running inside a Midnight Commander shell. You may get out-of-memory error messages when running the tests. These are normal and can be safely ignored.

Command Explanations

`--disable-doxygen-docs`: This switch disables doxygen documentation build and install, if you have doxygen installed. If doxygen is installed, and you wish to build them, remove this parameter.

`--disable-xml-docs`: This switch disables html documentation build and install, if you have xmlio installed. If xmlio is installed, and you wish to build them, remove this parameter.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-user-session`: This parameter enables per-user DBus sessions with `systemd`.

`--with-system-socket=/run/dbus/system_bus_socket`: This parameter specifies the location of the system bus socket.

`--enable-tests`: Builds extra parts of the code to support all tests. Do not use on a production build.

`--enable-embedded-tests`: Builds extra parts of the code to support only unit tests. Do not use on a production build.

`--enable-asserts`: Enables debugging code to run assertions for statements normally assumed to be true. This prevents a warning that '`--enable-tests`' on its own is only useful for profiling and might not give true results for all tests, but adds its own NOTE that this should not be used in a production build.

Configuring D-Bus

Config Files

```
/etc/dbus-1/session.conf, /etc/dbus-1/system.conf and /etc/dbus-1/system.d/*
```

Configuration Information

The configuration files listed above should probably not be modified. If changes are required, you should create `/etc/dbus-1/session-local.conf` and/or `/etc/dbus-1/system-local.conf` and make any desired changes to these files.

If any packages install a D-Bus `.service` file outside of the standard `/usr/share/dbus-1/services` directory, that directory should be added to the local session configuration. For instance, `/usr/local/share/dbus-1/services` can be added by performing the following commands as the `root` user:

```
cat > /etc/dbus-1/session-local.conf << "EOF"
<!DOCTYPE busconfig PUBLIC
"-//freedesktop//DTD D-BUS Bus Configuration 1.0//EN"
"http://www.freedesktop.org/standards/dbus/1.0/busconfig.dtd">
<busconfig>

<!-- Search for .service files in /usr/local -->
<servicedir>/usr/local/share/dbus-1/services</servicedir>

</busconfig>
EOF
```

D-Bus Session Daemon

There are many methods you can use to start a session daemon using the `dbus-launch` command. Review the `dbus-launch` man page for details about the available parameters and options. Here are some suggestions and examples:

- Add `dbus-launch` to the line in the `~/.xinitrc` file that starts your graphical desktop environment.
- If you use `gdm` or some other display manager that calls the `~/.xsession` file, you can add `dbus-launch` to the line in your `~/.xsession` file that starts your graphical desktop environment. The syntax would be similar to the example in the `~/.xinitrc` file.
- The examples shown previously use `dbus-launch` to specify a program to be run. This has the benefit (when also using the `-exit-with-x11` parameter) of stopping the session daemon when the specified program is stopped. You can also start the session daemon in your system or personal startup scripts by adding the following lines:

```
# Start the D-Bus session daemon
eval `dbus-launch`
export DBUS_SESSION_BUS_ADDRESS
```

This method will not stop the session daemon when you exit your shell, therefore you should add the following line to your `~/.bash_logout` file:

```
# Kill the D-Bus session daemon
kill $DBUS_SESSION_BUS_PID
```

Contents

A list of the installed files, along with their short descriptions can be found at [..../.../.../lfs/view/12.2-systemd/chapter08/dbus.html#contents-dbus](#).

Fcron-3.2.1

Introduction to Fcron

The Fcron package contains a periodical command scheduler which aims at replacing Vixie Cron.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://fcron.free.fr/archives/fcron-3.2.1.src.tar.gz>
- Download MD5 sum: bd4996e941a40327d11efc5e3fd1f839
- Download size: 587 KB
- Estimated disk space required: 5.1 MB
- Estimated build time: 0.1 SBU

Fcron Dependencies

Optional

An [MTA](#), [text editor](#) (default is `vi` from the [Vim-9.1.0660](#) package), [Linux-PAM-1.6.1](#), and [DocBook-utils-0.6.14](#)

Installation of Fcron

For security reasons, an unprivileged user and group for Fcron should be created (perform as the `root` user):

```
groupadd -g 22 fcron &&
useradd -d /dev/null -c "Fcron User" -g fcron -s /bin/false -u 22 fcron
```

Now fix some locations hard coded in the documentation:

```
find doc -type f -exec sed -i 's:/usr/local::g' {} \;
```

Install Fcron by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc      \
            --localstatedir=/var      \
            --without-sendmail      \
            --with-piddir=/run      \
            --with-boot-install=no &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

DESTDIR install must be done as `root` user. Furthermore, if PAM configuration files should be installed in `/etc/pam.d`, you have to create this directory in the DESTDIR before doing the install.

Command Explanations

`--without-sendmail`: By default, Fcron will attempt to use the `sendmail` command from an [MTA](#) package to email you the results of the `fcron` script. This switch is used to disable default email notification. Omit the switch to enable the default. Alternatively, you can use the `--with-sendmail=</path/to/MTA command>` to use a different mailer command.

`--with-boot-install=no`: This prevents installation of the bootscript included with the package.

`--with-piddir=/run`: This fixes the systemd unit to use the proper directory for PID files so that systemctl does not hang and the unit starts correctly.

`--with-editor=</path/to/editor>`: This switch allows you to set the default text editor.

`--with-dsssl-dir=</path/to/dsssl-stylesheets>`: May be used if you have [DocBook-utils-0.6.14](#) installed. Currently, the dsssl stylesheets are located at `/usr/share/sgml/docbook/dsssl-stylesheets-1.79`.

Configuring Fcron

Config Files

`/etc/fcron.conf`, `/etc/fcron.allow`, and `/etc/fcron.deny`

Configuration Information

There are no required changes in any of the config files. Configuration information can be found in the man page for `fcron.conf`.

`fcron` scripts are written using `fcrontab`. Refer to the `fcrontab` man page for proper parameters to address your situation.

If Linux-PAM is installed, two PAM configuration files are installed in `/etc/pam.d`. Alternatively if `/etc/pam.d` is not used, the installation will append two configuration sections to the existing `/etc/pam.conf` file. You should ensure the files match your preferences. Modify them as required to suit your needs.

Periodic Jobs

If you would like to set up a periodic hierarchy for the root user, first issue the following commands (as the `root` user) to create the `/usr/bin/run-parts` script:

```
cat > /usr/bin/run-parts << "EOF" &&
#!/bin/sh
# run-parts: Runs all the scripts found in a directory.
# from Slackware, by Patrick J. Volkerding with ideas borrowed
# from the Red Hat and Debian versions of this utility.

# keep going when something fails
set +e

if [ $# -lt 1 ]; then
    echo "Usage: run-parts <directory>"
    exit 1
fi

if [ ! -d $1 ]; then
    echo "Not a directory: $1"
    echo "Usage: run-parts <directory>"
    exit 1
fi

# There are several types of files that we would like to
# ignore automatically, as they are likely to be backups
# of other scripts:
IGNORE_SUFFIXES=~ ^ .bak .new .rpmsave .rpmorig .rpmnew .swp"

# Main loop:
for SCRIPT in $1/* ; do
    # If this is not a regular file, skip it:
    if [ ! -f $SCRIPT ] ; then
        continue
    fi
    # Determine if this file should be skipped by suffix:
    SKIP=false
    for SUFFIX in $IGNORE_SUFFIXES ; do
        if [ ! "$basename $SCRIPT $SUFFIX" = "$basename $SCRIPT" ] ; then
            SKIP=true
            break
        fi
    done
    if [ "$SKIP" = "true" ] ; then
        continue
    fi
    # If we've made it this far, then run the script if it's executable:
    if [ -x $SCRIPT ] ; then
        $SCRIPT || echo "$SCRIPT failed."
    fi
done

exit 0
EOF
chmod -v 755 /usr/bin/run-parts
```

Next, create the directory layout for the periodic jobs (again as the `root` user):

```
install -vdm754 /etc/cron.{hourly,daily,weekly,monthly}
```

Finally, add the `run-parts` to the system fcrontab (while still the `root` user):

```
cat > /var/spool/fcron/systab.orig << "EOF"
&bootrun 01 * * * * root run-parts /etc/cron.hourly
&bootrun 02 4 * * * root run-parts /etc/cron.daily
&bootrun 22 4 * * 0 root run-parts /etc/cron.weekly
&bootrun 42 4 1 * * root run-parts /etc/cron.monthly
EOF
```

Systemd Unit

Enable fcron to start at boot using the previously installed systemd unit.

```
systemctl enable fcron
```

Finally, again as the `root` user, start fcron and generate the `/var/spool/fcron/systab` file:

```
systemctl start fcron &&  
fcrontab -z -u systab
```

Contents

Installed Programs: `fcron`, `fcrondyn`, `fcronsighup`, and `fcrontab`

Installed Libraries: None

Installed Directories: `/usr/share/doc/fcron-3.2.1` and `/var/spool/fcron`

Short Descriptions

<code>fcron</code>	is the scheduling daemon
<code>fcrondyn</code>	is a user tool intended to interact with a running <code>fcron</code> daemon
<code>fcronsighup</code>	instructs <code>fcron</code> to reread the Fcron tables
<code>fcrontab</code>	is a program used to install, edit, list and remove the tables used by <code>fcron</code>

GPM-1.20.7

Introduction to GPM

The GPM (General Purpose Mouse daemon) package contains a mouse server for the console and `xterm`. It not only provides cut and paste support generally, but its library component is used by various software such as Links to provide mouse support to the application. It is useful on desktops, especially if following (Beyond) Linux From Scratch instructions; it's often much easier (and less error prone) to cut and paste between two console windows than to type everything by hand!

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/gpm/gpm-1.20.7.tar.bz2>
- Download MD5 sum: bf84143905a6a903dbd4d4b911a2a2b8
- Download size: 820 KB
- Estimated disk space required: 7.4 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/gpm-1.20.7-consolidated-1.patch>

GPM Dependencies

Optional

`texlive-20240312` (for documentation)

Kernel Configuration

Enable the following option in the kernel configuration and recompile the kernel if necessary:

```
Device Drivers --->  
  Input device support --->  
    -*- Generic input layer (needed for keyboard, mouse, ...)           [INPUT]  
    <*/M>  Mouse interface                                         [INPUT_MOUSEDEV]
```

Installation of GPM

Install GPM by running the following commands:

```

patch -Np1 -i ../gpm-1.20.7-consolidated-1.patch
./autogen.sh
./configure --prefix=/usr --sysconfdir=/etc ac_cv_path_emacs=no &&
make

```

If [texlive-20240312](#) is installed, build the GPM manual as dvi, ps, and pdf formats:

```

make -C doc gpm.{dvi,ps} &&
dvipdfm doc/gpm.dvi -o doc/gpm.pdf

```

This package does not come with a test suite.

Now, as the `root` user:

```

make install &&

install-info --dir-file=/usr/share/info/dir \
              /usr/share/info/gpm.info &&

rm -fv /usr/lib/libgpm.a &&
ln -sfv libgpm.so.2.1.0 /usr/lib/libgpm.so &&
install -v -m644 conf/gpm-root.conf /etc &&

install -v -m755 -d /usr/share/doc/gpm-1.20.7/support &&
install -v -m644 doc/support/* \
              /usr/share/doc/gpm-1.20.7/support &&
install -v -m644 doc/{FAQ,HACK_GPM,README*} \
              /usr/share/doc/gpm-1.20.7

```

If [texlive-20240312](#) is installed and you've built GPM manual as dvi, ps, and pdf formats, install them as the `root` user:

```

install -vm644 doc/gpm.{dvi,ps,pdf} /usr/share/doc/gpm-1.20.7

```

Command Explanations

`./autogen.sh`: This command creates the missing `configure` script.

`ac_cv_path_emacs=no`: This variable works around an issue causing the package fail to build with [Emacs-29.4](#) installed. It also suppresses the installations of some "Emacs support files" shipped with GPM. These files are quite outdated and they should be superseded with [Emacs-29.4](#) built-in GPM support. If you need to use [Emacs-29.4](#) in Linux console with mouse support, you should install (or reinstall) [Emacs-29.4](#) after GPM.

`install-info ...`: This package installs a `.info` file, but does not update the system `dir` file. This command makes the update.

`ln -v -sfv libgpm.so.2.1.0 /usr/lib/libgpm.so`: This command is used to create (or update) the `.so` symlink to the library.

Configuring GPM

Systemd Unit

To start the `gpm` daemon at boot, install the systemd unit from the [blfs-systemd-units-20240801](#) package by running the following command as the `root` user:

```
make install-gpm
```

Config Files

`/etc/gpm-root.conf` and `~/.gpm-root`: The default and individual user `gpm-root` configuration files.

Configuration Information

GPM is by default started with the following parameters: `-m /dev/input/mice -t imps2`. If the mentioned parameters don't suit your needs, you can override them by running the following commands as the `root` user:

```

install -v -dm755 /etc/systemd/system/gpm.service.d &&
cat > /etc/systemd/system/gpm.service.d/99-user.conf << EOF
[Service]

```

```
ExecStart=/usr/sbin/gpm <list of parameters>
EOF
```

Contents

- Installed Programs:** disable-paste, display-buttons, display-coords, get-versions, gpm, gpm-root, hittest, mev, and mouse-test
Installed Library: libgpm.so
Installed Directory: /usr/share/doc/gpm-1.20.7

Short Descriptions

disable-paste	is a security mechanism used to disable the paste buffer
display-buttons	is a simple program that reports the mouse buttons being pressed and released
display-coords	is a simple program that reports the mouse coordinates
get-versions	is used to report the GPM library and server versions
gpm	is a cut and paste utility and mouse server for virtual consoles
gpm-root	is a default handler for gpm. It is used to draw menus on the root window
hittest	is a simple sample application using the high-level library, meant to be read by programmers trying to use the high-level library
mev	is a program to report mouse events
mouse-test	is a tool for determining the mouse type and device it's attached to
libgpm.so	contains the API functions to access the GPM daemon

Hdparm-9.65

Introduction to Hdparm

The Hdparm package contains a utility that is useful for obtaining information about, and controlling ATA/IDE controllers and hard drives. It allows to increase performance and sometimes to increase stability.

This package is known to build and work properly using an LFS 12.2 platform.

Warning

As well as being useful, incorrect usage of Hdparm can destroy your information and in rare cases, drives. Use with caution and make sure you know what you are doing. If in doubt, it is recommended that you leave the default kernel parameters alone.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/hdparm/hdparm-9.65.tar.gz>
- Download MD5 sum: 6d6d039d61ec995b1ec72ddce0b1853b
- Download size: 140 KB
- Estimated disk space required: 1.0 MB
- Estimated build time: less than 0.1 SBU

Installation of Hdparm

Build Hdparm by running the following command:

```
make
```

This package does not come with a test suite.

Now, as the root user:

```
make binprefix=/usr install
```

Contents

Installed Program: hparm

Installed Libraries: None

Installed Directories: None

Short Descriptions

hparm provides a command-line interface to various hard disk ioctls supported by the stock Linux ATA/IDE device driver subsystem

hwdata-0.385

Introduction to hwdata

The hwdata package contains current PCI and vendor id data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/vcrhonek/hwdata/archive/v0.385/hwdata-0.385.tar.gz>
- Download MD5 sum: 841090258addcaf8841f78f75eeaaadd
- Download size: 2.3 MB
- Estimated disk space required: 9.4 MB
- Estimated build time: less than 0.1 SBU

Installation of hwdata

Install hwdata by running the following commands:

```
./configure --prefix=/usr --disable-blacklist
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: None

Installed Directory: /usr/share/hwdata

LSB-Tools-0.12

Introduction to LSB-Tools

The LSB-Tools package includes tools for Linux Standards Base (LSB) conformance.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lfs-book/LSB-Tools/releases/download/v0.12/LSB-Tools-0.12.tar.gz>
- Download MD5 sum: 1e6ef8cdfddb55035a6c36757e6313f9
- Download size: 20 KB
- Estimated disk space required: 412 KB
- Estimated build time: less than 0.1 SBU

Installation of LSB-Tools

Install LSB-Tools by running the following commands:

```
make
```

Now, as the `root` user:

```
make install
```

Remove a file that should not be installed as the `root` user:

```
rm /usr/sbin/lsbinstall
```

Remove two scripts that serve no purpose on a systemd system:

```
rm /usr/sbin/{install,remove}_initd
```

Configuration Information

The configuration for this package was done in [LFS](#). The file `/etc/lsb-release` should already exist. Be sure that the `DISTRIB_CODENAME` entry has been set properly.

Contents

Installed Programs: `lsb_release`

Installed Library: None

Installed Directories: `/usr/lib/lsb` and `/usr/lib/python3.12/site-packages/lsbtools`

Short Descriptions

`lsb_release` is a script to give LSB data

Lm-sensors-3-6-0

Introduction to Lm_sensors

The `lm_sensors` package provides userspace support for the hardware monitoring drivers in the Linux kernel. This is useful for monitoring the temperature of the CPU and adjusting the performance of some hardware (such as cooling fans).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lm-sensors/lm-sensors/archive/V3-6-0/lm-sensors-3-6-0.tar.gz>
- Download MD5 sum: f60e47b5eb50bbeed48a9f43bb08dd5e
- Download size: 268 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: less than 0.1 SBU

lm_sensors Dependencies

Required

[Which-2.21](#)

Optional

[RRDtool](#) (required to build the `sensord` program) and [dmidecode](#) (runtime)

Kernel Configuration

The following configuration options attempt to cover the most common hardware monitoring devices in a typical desktop or laptop system. View the help of each (by pressing the `H` button with the option focused in the `make menuconfig`) to know if you need it. There are many platform-specific hardware monitoring devices so it's impossible to list the configuration for all

of them here. You may investigate the content of `/sys/class/hwmon` within a “mainstream” distro running on the system to know which drivers you need.

```
Power management and ACPI options --->
[*] ACPI (Advanced Configuration and Power Interface) Support ---> [ACPI]
  < /*/M> Battery [ACPI_BATTERY]
  < /*/M> Thermal Zone [ACPI_THERMAL]

Device Drivers --->
NVME Support --->
  < /*> NVMe Express block device [BLK_DEV_NVME]
  # Set [HWMON] to <*> (not <M>) or it will not show up:
  [ /*] NVMe hardware monitoring [NVME_HWMON]
<*/M> Hardware Monitoring support --->
  < /*/M> AMD Athlon64/FX or Opteron temperature sensor [SENSORS_K8TEMP]
  < /*/M> AMD Family 10h+ temperature sensor [SENSORS_K10TEMP]
  < /*/M> AMD Family 15h processor power [SENSORS_FAM15H_POWER]
  < /*/M> Intel Core/Core2/Atom temperature sensor [SENSORS_CORETEMP]
```

Recompile your kernel and reboot into the new kernel.

Installation of Lm_sensors

Install Lm_sensors by running the following commands:

```
make PREFIX=/usr \
BUILD_STATIC_LIB=0 \
MANDIR=/usr/share/man \
EXLDFLAGS=
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr \
BUILD_STATIC_LIB=0 \
MANDIR=/usr/share/man install &&

install -v -m755 -d /usr/share/doc/lm_sensors-3-6-0 &&
cp -rv README INSTALL doc/* \
/usr/share/doc/lm_sensors-3-6-0
```

Command Explanations

`BUILD_STATIC_LIB=0`: This parameter disables compiling and installing the static version of `libsensors`.

`EXLDFLAGS=`: This parameter disables hard coding library search paths (rpath) into the binary executable files and shared libraries. This package does not need rpath for an installation into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`PROG_EXTRA=sensord`: This parameter enables compiling `sensord`, a daemon that can monitor your system at regular intervals. Compiling `sensord` requires [RRDtool](#). Be sure to install RRDtool in `/usr` by running `make prefix=/usr` when building it. Otherwise, Lm_sensors will not find it easily.

Configuring Lm_sensors

Config File

`/etc/sensors3.conf`

Contents

Installed Programs: `fancontrol`, `isadump`, `isaset`, `pwmconfig`, `sensors`, `sensors-conf-convert`, `sensors-detect`, and optionally, `sensord`

Installed Library: `libsensors.so`

Installed Directories: `/etc/sensors.d`, `/usr/include/sensors` and `/usr/share/doc/lm_sensors-3-6-0`

Short Descriptions

<code>fancontrol</code>	is a shell script for use with <code>lm_sensors</code> . It reads its configuration from a file, then calculates fan speeds from temperatures and sets the corresponding PWM outputs to the computed values
<code>isadump</code>	is a small helper program to examine registers visible through the ISA bus. It is intended to probe any chip that lives on the ISA bus working with an address register and a data register (I2C-like access) or a flat range (of up to 256 bytes)
<code>isaset</code>	is a small helper program to set registers visible through the ISA bus
<code>pwmconfig</code>	tests the pulse width modulation (PWM) outputs of sensors and configures fancontrol
<code>sensors</code>	prints the current readings of all sensor chips
<code>sensors-conf-convert</code>	is a Perl script to convert <code>lm-sensors</code> version 2 configuration files to work with version 3
<code>sensors-detect</code>	is a Perl script that will walk you through the process of scanning your system for various hardware monitoring chips (<code>sensors</code>) supported by <code>libsensors</code> , or more generally by the <code>lm_sensors</code> tool suite
<code>sensord</code>	(optional) is a daemon that can be used to periodically log sensor readings
<code>libsensors.so</code>	contains the <code>lm_sensors</code> API functions

Logrotate-3.22.0

Introduction to Logrotate

The logrotate package allows automatic rotation, compression, removal, and mailing of log files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/logrotate/logrotate/releases/download/3.22.0/logrotate-3.22.0.tar.xz>
- Download MD5 sum: 2386501a53ff086f44eeada2b27d50b8
- Download size: 172 KB
- Estimated disk space required: 2.6 MB (add 38 MB for tests)
- Estimated build time: less than 0.1 SBU (add 0.1 SBU for tests)

Logrotate Dependencies

Required

[popt-1.19](#)

Optional

An [MTA](#) (runtime)

Installation of Logrotate

Install logrotate by running the following command:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make test`. Two tests fail if an MTA is not installed.

Now, as the `root` user:

```
make install
```

Configuring Logrotate

Logrotate needs a configuration file, which must be passed as an argument to the command when executed. Create the file as the `root` user:

```
cat > /etc/logrotate.conf << EOF
# Begin /etc/logrotate.conf
```

```

# Rotate log files weekly
weekly

# Don't mail logs to anybody
nomail

# If the log file is empty, it will not be rotated
notifempty

# Number of backups that will be kept
# This will keep the 2 newest backups only
rotate 2

# Create new empty files after rotating old ones
# This will create empty log files, with owner
# set to root, group set to sys, and permissions 664
create 0664 root sys

# Compress the backups with gzip
compress

# No packages own lastlog or wtmp -- rotate them here
/var/log/wtmp {
    monthly
    create 0664 root utmp
    rotate 1
}

/var/log/lastlog {
    monthly
    rotate 1
}

# Some packages drop log rotation info in this directory
# so we include any file in it.
include /etc/logrotate.d

# End /etc/logrotate.conf
EOF

chmod -v 0644 /etc/logrotate.conf

```

Now create the `/etc/logrotate.d` directory as the `root` user:

```
mkdir -p /etc/logrotate.d
```

At this point additional log rotation commands can be entered, typically in the `/etc/logrotate.d` directory. For example:

```

cat > /etc/logrotate.d/sys.log << EOF
/var/log/sys.log {
    # If the log file is larger than 100kb, rotate it
    size 100k
    rotate 5
    weekly
    postrotate
        /bin/killall -HUP syslogd
    endscript
}
EOF

chmod -v 0644 /etc/logrotate.d/sys.log

```

You can designate multiple files in one entry:

```

cat > /etc/logrotate.d/example.log << EOF
file1
file2
file3 {
    ...
    postrotate
    ...
    endscript
}
EOF

```

```
chmod -v 0644 /etc/logrotate.d/example.log
```

You can use in the same line the list of files: file1 file2 file3. See the logrotate man page or <https://www.techrepublic.com/article/manage-linux-log-files-with-logrotate/> for more examples.

The command `logrotate /etc/logrotate.conf` can be run manually, however, the command should be run daily. Other useful commands are `logrotate -d /etc/logrotate.conf` for debugging purposes and `logrotate -f /etc/logrotate.conf` forcing the logrotate commands to be run immediately. Combining the previous options `-df`, you can debug the effect of the force command. When debugging, the commands are only simulated and are not actually run. As a result, errors about files not existing will eventually appear because the files are not actually created.

To run the `logrotate` command daily, execute the following commands, as the `root` user, to create a `systemd` timer to run daily at 3:00 A.M. (local time):

```
cat > /usr/lib/systemd/system/logrotate.service << "EOF" &&
[Unit]
Description=Runs the logrotate command
Documentation=man:logrotate(8)
DefaultDependencies=no
After=local-fs.target
Before=shutdown.target

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/usr/sbin/logrotate /etc/logrotate.conf
EOF
cat > /usr/lib/systemd/system/logrotate.timer << "EOF" &&
[Unit]
Description=Runs the logrotate command daily at 3:00 AM

[Timer]
OnCalendar=*-*-* 3:00:00
Persistent=true

[Install]
WantedBy=timers.target
EOF
systemctl enable logrotate.timer
```

Contents

Installed Programs: logrotate

Installed Library: None

Installed Directories: None

Short Descriptions

`logrotate` performs the log maintenance functions defined in the configuration files

MC-4.8.32

Introduction to MC

MC (Midnight Commander) is a text-mode full-screen file manager and visual shell. It provides a clear, user-friendly, and somewhat protected interface to a Unix system while making many frequent file operations more efficient and preserving the full power of the command prompt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://ftp.midnight-commander.org/mc-4.8.32.tar.xz>
- Download MD5 sum: bcc9043a815c02c5837f8b1a4a60da5a
- Download size: 2.3 MB
- Estimated disk space required: 71 MB (add 96 MB for tests)
- Estimated build time: 0.3 SBU (using parallelism=4; add 0.2 SBU for tests)

MC Dependencies

Required

[GLib-2.80.4](#)

Recommended

[slang-2.3.3](#)

Optional

[Doxygen-1.12.0](#), [GPM-1.20.7](#), [Graphviz-12.1.0](#), [libssh2-1.11.0](#), [pcre2-10.44](#), [Ruby-3.3.4](#), [UnZip-6.0](#), a graphical environment, and [Zip-3.0](#)

Installation of MC

Install MC by running the following commands:

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --enable-charset &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--sysconfdir=/etc`: This switch places the global configuration directory in `/etc`.

`--enable-charset`: This switch adds support to `mcedit` for editing files in encodings different from the one implied by the current locale.

`--with-screen=ncurses`: Use this if you don't have [slang-2.3.3](#) installed.

`--with-search-engine=pcre2`: Use this switch if you would prefer to use [pcre2-10.44](#) instead of GLib for the built-in search engine.

Configuring MC

Config Files

`~/.config/mc/*`

Configuration Information

The `~/.config/mc` directory and its contents are created when you start `mc` for the first time. Then you can edit the main `~/.config/mc.ini` configuration file manually or through the MC shell. Consult the [mc\(1\)](#) man page for details.

Contents

Installed Programs: `mc` and the symlinks `mcdiff`, `mcedit` and `mcview`

Installed Libraries: None

Installed Directories: `/etc/mc` and `/usr/{libexec,share}/mc`

Short Descriptions

<code>mc</code>	is a visual shell
<code>mcdiff</code>	is an internal visual diff tool
<code>mcedit</code>	is an internal file editor
<code>mcview</code>	is an internal file viewer

ModemManager-1.18.12

Introduction to ModemManager

ModemManager provides a unified high level API for communicating with mobile broadband modems, regardless of the protocol used to communicate with the actual device.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/ModemManager/ModemManager-1.18.12.tar.xz>
 - Download MD5 sum: 9f014dfc59f1bd8bc230bb2c2974d104
 - Download size: 2.5 MB
 - Estimated disk space required: 155 MB (with tests)
 - Estimated build time: 0.7 SBU (Using parallelism=4; with tests)

ModemManager Dependencies

Required

libqudev-238

Recommended

Glib-2.80.4 (with GObject Introspection), **libmbim-1.26.4**, **libqmi-1.30.8**, **Polkit-125**, and **Vala-0.56.17**

Optional

[GTK-Doc-1.34.0](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/ModemManager>

Installation of ModemManager

Install ModemManager by running the following commands:

```
./configure --prefix=/usr  
          --sysconfdir=/etc  
          --localstatedir=/var  
          --disable-static  
          --disable-maintainer-mode  
          --with-systemd-journal  
          --with-systemd-suspend-resume &&  
  
make
```

To test the results, issue: `make check`.

Now, as the *root* user:

`make install`

Command Explanations

`--with-systemd-suspend-resume`: This switch forces ModemManager to use the `systemd` power management facilities.

`--with-systemd-journal`: This switch forces ModemManager to use the systemd journal for logging.

--disable-static: This switch prevents installation of static versions of the libraries.

--enable-gtk-doc: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Configuring ModemManager

Systemd Units

To start the `ModemManager` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable ModemManager
```

Contents

Installed Programs: mmcli and ModemManager

Installed Libraries:

Installed Directories: /etc/ModemManager, /usr/include/libmm-glib, /usr/include/ModemManager, /usr/lib/ModemManager, /usr/share/ModemManager, /usr/share/gtk-doc/html/libmm-glib (optional), and /usr/share/gtk-doc/html/ModemManager (optional)

Short Descriptions

mmcli	is an utility used to control and monitor the ModemManager
ModemManager	is a D-Bus service used to communicate with modems
libmm- glib.so	contains API functions for communicating with mobile broadband modems, regardless of the protocol used to communicate with the actual device

notification-daemon-3.20.0

Introduction to Notification Daemon

The Notification Daemon package contains a daemon that displays passive pop-up notifications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/notification-daemon/3.20/notification-daemon-3.20.0.tar.xz>
 - Download MD5 sum: 2de7f4075352831f1d98d8851b642124
 - Download size: 336 KB
 - Estimated disk space required: 4.1 MB
 - Estimated build time: less than 0.1 SBU

Notification Daemon Dependencies

Required

GTK+-3.24.43 and libcanberra-0.30 (Built with [GTK+-3.24.43](#) support).

Installation of Notification Daemon

Install Notification Daemon by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --disable-static  &&
make
```

This package does not come with a test suite.

Now, as the *root* user:

make install

You can test the notification daemon with the command `notify-send`:

```
pgrep -l notification-da &&
notify-send -i info Information "Hi ${USER}. This is a Test"
```

The command `pgrep -l notification-da` is added to assure that it is the daemon of this package that is running, not another one, e.g. the daemon from [xfce4-notifyd-0.9.6](#).

Contents

Installed Program:	none
Installed Libraries:	none
Installed Directory:	none

p7zip-17.04

Introduction to p7zip

p7zip is the Unix command-line port of 7-Zip, a file archiver that archives with high compression ratios. It handles 7z, ZIP, GZIP, Brotli, BZIP2, XZ, TAR, APM, ARJ, CAB, CHM, CPIO, CramFS, DEB, DMG, FAT, HFS, ISO, Lizard, LZ5, LZFSE, LZH, LZMA, LZMA2, MBR, MSI, MSLZ, NSIS, NTFS, RAR, RPM, SquashFS, UDF, VHD, WIM, XAR, Z, and Zstd formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/p7zip-project/p7zip/archive/v17.04/p7zip-17.04.tar.gz>
- Download MD5 sum: 00acf6be87848231722d2d53f89e4a5
- Download size: 6.5 MB
- Estimated disk space required: 55 MB
- Estimated build time: 2.3 SBU (with tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/p7zip-17.04-consolidated_fixes-1.patch

p7zip Dependencies

Optional

[wxWidgets](#)

Installation of p7zip

First, fix several security vulnerabilities:

```
patch -Np1 -i ../p7zip-17.04-consolidated_fixes-1.patch
```

Next, prevent p7zip from installing compressed manual pages:

```
sed '/^gzip/d' -i install.sh
```

Install p7zip by running the following commands:

```
make all3
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make DEST_HOME=/usr \
DEST_MAN=/usr/share/man \
DEST_SHARE_DOC=/usr/share/doc/p7zip-17.04 install
```

Note

If using DESTDIR techniques, use DEST_DIR instead.

Contents

Installed Programs: 7z, 7za, and 7zr

Installed Libraries: None

Installed Directory: /usr/lib/p7zip and /usr/share/doc/p7zip-17.04

Short Descriptions

- 7z is a file archiver utility
- 7za is a stand-alone executable handling less archive formats than 7z
- 7zz is a minimal version of 7za that handles only 7z archives

Pax-20240817

Introduction to Pax

pax is an archiving utility created by POSIX and defined by the POSIX.1-2001 standard. Rather than sort out the incompatible options that have crept up between tar and cpio, along with their implementations across various versions of UNIX, the IEEE designed a new archive utility. The name "pax" is an acronym for portable archive exchange. Furthermore, "pax" means "peace" in Latin, so its name implies that it shall create peace between the tar and cpio format supporters. The command invocation and command structure is somewhat a unification of both tar and cpio.

pax has been required to be present in LSB conformant systems since LSB version 3.0.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://www.mirbsd.org/MirOS/dist/mir/cpio/paxmirabilis-20240817.tgz>
- Download MD5 sum: 9a723154a4201a0892b7ff815b6753b5
- Download size: 180 KB
- Estimated disk space required: 1.6 MB
- Estimated build time: less than 0.1 SBU

Installation of Pax

Install pax by running the following commands::

Note

This package expands to the directory *pax*.

```
bash Build.sh
```

This package does not come with a test suite.

Now, as the *root* user:

Note

This package also creates hard links from *pax* to the programs *cpio* and *tar* in the build directory. The LFS editors do not recommend that they be installed as they will overwrite the GNU versions of these programs.

```
install -v pax /usr/bin &&
install -v -m644 pax.1 /usr/share/man/man1
```

Contents

Installed Program: pax

Short Descriptions

pax copies files to and from archives in several formats

pciutils-3.13.0

Introduction to PCI Utils

The PCI Utils package contains a set of programs for listing PCI devices, inspecting their status and setting their configuration registers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://mj.ucw.cz/download/linux/pci/pciutils-3.13.0.tar.gz>
- Download MD5 sum: 1edb865de7a2de84e67508911010091b
- Download size: 660 KB
- Estimated disk space required: 4.9 MB
- Estimated build time: less than 0.1 SBU

pciutils Dependencies

Recommended

[hwdata-0.385](#) (runtime)

Installation of PCI Utils

Prevent the installation of the `pci.ids` file to avoid a conflict with the [hwdata-0.385](#) package:

```
sed -r '/INSTALL/{/PCI_IDS|update-pciids /d; s/update-pciids.8//}' \
-i Makefile
```

Install PCI Utils by running the following commands:

```
make PREFIX=/usr \
SHAREDIR=/usr/share/hwdata \
SHARED=yes
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr \
SHAREDIR=/usr/share/hwdata \
SHARED=yes \
install install-lib \
&&
chmod -v 755 /usr/lib/libpci.so
```

Next, install the [hwdata-0.385](#) package for the `pci.ids` file.

Command Explanations

`SHARED=yes`: This parameter enables building of the shared library instead of the static one.

Contents

Installed Programs: lspci, pcilmr, and setpci

Installed Library: libpci.so

Installed Directory: /usr/include/pci

Short Descriptions

<code>lspci</code>	is a utility for displaying information about all PCI buses in the system and all devices connected to them
<code>pcilmr</code>	is a utility for margining PCIe links
<code>setpci</code>	is a utility for querying and configuring PCI devices
<code>libpci.so</code>	is a library that allows applications to access the PCI subsystem

pm-utils-1.4.1

Introduction to Power Management Utilities

The Power Management Utilities provide simple shell command line tools to suspend and hibernate the computer. They can be used to run user supplied scripts on suspend and resume.

Note

On a system using systemd as the init system, systemd already provides this functionality. As a result, this package is probably not needed, and it may conflict with systemd. Read the documentation for `suspend.target` and `hibernate.target` in the man page [systemd.special\(7\)](#) for more details.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://pm-utils.freedesktop.org/releases/pm-utils-1.4.1.tar.gz>
- Download MD5 sum: 1742a556089c36c3a89eb1b957da5a60
- Download size: 204 KB
- Estimated disk space required: 1.6 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/pm-utils-1.4.1-bugfixes-1.patch>

Power Management Utilities Dependencies

Optional

[xmlto-0.0.29](#) (to generate man pages)

Optional (runtime)

[hdparm-9.65](#), [Wireless Tools-29](#), [ethtool](#), and [vbetool](#)

Kernel Configuration

If needed, enable the following options in the kernel configuration and recompile the kernel:

```
Power management and ACPI options --->
  [*] Suspend to RAM and standby           [SUSPEND]
  [*] Hibernation (aka 'suspend to disk')  [HIBERNATION]
```

Suspend to RAM allows the system to enter sleep states in which main memory is powered and thus its contents are preserved. The method cuts power to most parts of the machine aside from the RAM. Because of the large power savings, it is advisable for laptops to automatically enter this mode when the computer is running on batteries and the lid is closed (or the user is inactive for some time).

Suspend to disk (Hibernation) saves the machine's state into swap space and completely powers off the machine. When the machine is powered on, the state is restored. Until then, there is zero power consumption. Suspend to RAM and hibernation are normally appropriate for portable devices such as laptops, but can be used on workstations. The capability is not really appropriate for servers.

To use hibernation, the kernel parameter `resume=/dev/<swap_partition>` has to be used on the kernel command line (in `grub.cfg`). The swap partition should be at least the size of the physical RAM on the system.

Installation of Power Management Utilities

First, fix several bugs and a couple of incompatibilities with newer kernels:

```
patch -Np1 -i ../pm-utils-1.4.1-bugfixes-1.patch
```

Install Power Management Utilities by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --docdir=/usr/share/doc/pm-utils-1.4.1 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you don't have [xmlto-0.0.29](#) installed, copy pregenerated man pages, as the `root` user:

```
install -v -m644 man/*.1 /usr/share/man/man1 &&
install -v -m644 man/*.8 /usr/share/man/man8 &&
ln -sv pm-action.8 /usr/share/man/man8/pm-suspend.8 &&
ln -sv pm-action.8 /usr/share/man/man8/pm-hibernate.8 &&
ln -sv pm-action.8 /usr/share/man/man8/pm-suspend-hybrid.8
```

Configuring Power Management Utilities

Suspend or resume functionality can be easily modified by installing files into the `/etc/pm/sleep.d` directory. These files, known as hooks, are run when the system is put into a sleep state or resumed. Default hooks are located in `/usr/lib/pm-utils/sleep.d`, and user hooks should be put in `/etc/pm/sleep.d`. See the [pm-action\(8\)](#) man page for more information.

In order to use hibernation with GRUB and a swap partition, you need to add kernel parameter `resume=swap_partition` (e.g. `resume=/dev/sda1`) to the kernel line in the `/boot/grub/grub.cfg` configuration file.

Contents

Installed Programs: `on_ac_power`, `pm-hibernate`, `pm-is-supported`, `pm-powersave`, `pm-suspend` and `pm-suspend-hybrid`
Installed Libraries: None
Installed Directories: `/etc/pm`, `/usr/lib/pm-utils` and `/usr/share/doc/pm-utils-1.4.1`

Short Descriptions

<code>on_ac_power</code>	is a script that determines whether the system is running on AC power (rather than a battery)
<code>pm-hibernate</code>	is a symlink to pm-action script that puts the computer into hibernate mode (the system is fully powered off and system state is saved to disk)
<code>pm-is-supported</code>	is a script that checks whether power management features such as suspend and hibernate are supported
<code>pm-powersave</code>	is a script that puts the computer into powersaving (low power) mode
<code>pm-suspend</code>	is a symlink to pm-action script that puts the computer into suspend mode (most devices are shut down and system state is saved in RAM)
<code>pm-suspend-hybrid</code>	is a symlink to pm-action script that puts the computer into hybrid-suspend mode (the system does everything it needs to hibernate, but suspends instead of shutting down)

Power-profiles-daemon-0.21

Introduction to Power-profiles-daemon

The Power-profiles-daemon package provides a program that allows modification of the system power/behavior state. This is used on many laptops and can be used by a Desktop Environment to activate power saving or performance CPU

governors through dbus. On other systems, Power-profiles-daemon can be used as a streamlined way to set the CPU governor in order to increase system performance at the cost of energy usage.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/upower/power-profiles-daemon/-/archive/0.21/power-profiles-daemon-0.21.tar.gz>
- Download MD5 sum: 0196aae5798263d3d8f33a9280a6cb10
- Download size: 76 KB
- Estimated disk space required: 3.2 MB
- Estimated build time: less than 0.1 SBU (with tests)

Power-profiles-daemon Dependencies

Required

[Polkit-125](#), [PyGObject-3.48.2](#) (pycairo is **not** needed), and [UPower-1.90.4](#)

Optional

[GTK-Doc-1.34.0](#), The rest are for the tests, [dbusmock-0.32.1](#), [umockdev-0.18.3](#), [isort](#), and [mccabe](#)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
Power management and ACPI options --->
CPU Frequency scaling --->
  --*-- CPU Frequency scaling                               [CPU_FREQ]
  --*-- 'performance' governor                           [CPU_FREQ_GOV_PERFORMANCE]
  <*/M>  'powersave' governor                          [CPU_FREQ_GOV_POWERSAVE]
  # Select if CPU is Intel:
  [/]* Intel P state control                         [X86_INTEL_PSTATE]
  # Select if CPU is AMD:
  [/]* AMD Processor P-State driver                  [X86_AMD_PSTATE]

Device Drivers --->
  # Some drivers under this submenu provide "platform profile" support
  # and power-profiles-daemon can take advantage from platform profiles;
  # select a driver if suitable for your platform:
  [/]* X86 Platform Specific Device Drivers --->      [X86_PLATFORM_DEVICES]
```

Select the appropriate sub-options that appear when the above options are selected. As much as possible, the layout should be the same as in kernel configuration menus.

Installation of Power-profiles-daemon

Install Power-profiles-daemon by running the following commands:

```
mkdir build &&
cd build &&

meson setup           \
  --prefix=/usr        \
  --buildtype=release  \
  -D gtk_doc=false    \
  -D tests=false       \
  .. &&
ninja
```

If you have installed the external dependencies, to test the results issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D gtk_doc=false`: Prevents building the documentation. Remove this if you have GTK-Doc installed and wish to build the documentation.

`-D tests=false`: Prevents building the tests because they cannot be run within the boundaries of BLFS. Remove this if you have installed the external dependencies and wish to run the tests.

Configuring Power-profiles-daemon

Systemd Unit

To start the power-profiles-daemon on boot, enable the systemd service that was installed by running the following command as the `root` user:

```
systemctl enable power-profiles-daemon
```

Select a Power Profile

To list all supported power profiles, issue:

```
powerprofilesctl
```

To activate a power profile (for example `performance`), issue:

```
powerprofilesctl set performance
```

Some desktop environments (for example GNOME and KDE) also provide a graphical interface to interact with `power-profiles-daemon` and activate a power profile.

The activated power profile is automatically stored in `/var/lib/power-profiles-daemon/state.ini` and `power-profiles-daemon` will read it out and activate it again on the next boot if the Systemd unit is enabled as above.

Contents

Installed Program: `powerprofilesctl`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`powerprofilesctl` This allows the user to set the power governor of the CPU.

Raptor-2.0.16

Introduction to Raptor

Raptor is a C library that provides a set of parsers and serializers that generate Resource Description Framework (RDF) triples.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.librdf.org/source/raptor2-2.0.16.tar.gz>
- Download MD5 sum: 0a71f13b6eaa0a04bf411083d89d7bc2
- Download size: 1.7 MB
- Estimated disk space required: 25 MB (additional 2 MB for the tests)
- Estimated build time: 0.1 SBU (additional 0.3 SBU for the tests)

Raptor Dependencies

Required

[cURL-8.9.1](#) and [libxslt-1.1.42](#)

Optional

[GTK-Doc-1.34.0](#), [ICU-75.1](#) and [libyaJL](#)

Installation of Raptor

First, fix an incompatibility with libxml2-2.11.x:

```
sed -i 's/20627/20627 \&\& LIBXML_VERSION < 21100/' src/raptor_libxml.c
```

Install Raptor by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

To test the results, issue: `make check`. Several of the XML tests may fail.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-icu-config=/usr/bin/icu-config`: Use this switch if you have installed [ICU-75.1](#) and wish to build Raptor with its support.

Contents

Installed Programs: rapper

Installed Libraries: libraptor2.so

Installed Directories: /usr/include/raptor2 and /usr/share/gtk-doc/html/raptor2

Short Descriptions

`rapper` is a RDF parsing and serializing utility
`libraptor2.so` contains the Raptor API functions

Rasql-0.9.33

Introduction to Rasql

Rasql is a C library that handles Resource Description Framework (RDF) query language syntaxes, query construction, and execution of queries returning results as bindings, boolean, RDF graphs/triples or syntaxes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.librdf.org/source/rasql-0.9.33.tar.gz>
- Download MD5 sum: 1f5def51ca0026cd192958ef07228b52
- Download size: 1.6 MB
- Estimated disk space required: 22 MB (additional 4 MB for the tests)
- Estimated build time: 0.3 SBU (additional 0.7 SBU for the tests)

Rasql Dependencies

Required

[Raptor-2.0.16](#)

Optional

[libgcrypt-1.11.0](#)

Installation of Rasqal

Install Rasqal by running the following commands:

```
./configure --prefix=/usr --disable-static &
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: `rasql-config` and `roqet`

Installed Library: `librasqal.so`

Installed Directories: `/usr/include/rasqal` and `/usr/share/gtk-doc/html/rasqal`

Short Descriptions

<code>rasql-config</code>	is an utility for retrieving the installation options of Rasqal
<code>roqet</code>	is an RDF query utility

Redland-1.0.17

Introduction to Redland

Redland is a set of free software C libraries that provide support for the Resource Description Framework (RDF).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.librdf.org/source/redland-1.0.17.tar.gz>
- Download MD5 sum: e5be03eda13ef68aabab6e42aa67715e
- Download size: 1.6 MB
- Estimated disk space required: 18 MB
- Estimated build time: 0.2 SBU

Redland Dependencies

Required

[Rasqal-0.9.33](#)

Optional

[SQLite-3.46.1](#), [MariaDB-10.11.8](#) or [MySQL](#), [PostgreSQL-16.4](#), [Berkeley DB](#) (deprecated) [libiodbc](#), [virtuoso](#), and [3store](#)

Installation of Redland

Install Redland by running the following commands:

```
./configure --prefix=/usr --disable-static &
make
```

To test the results, issue `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: rdfproc, redland-config and redland-db-upgrade

Installed Libraries: librdf.so and /usr/lib/redland/librdf_storage_*.so

Installed Directories: /usr/lib/redland, /usr/share/gtk-doc/html/redland and /usr/share/redland

Short Descriptions

<code>rdfproc</code>	is the Redland RDF processor utility
<code>redland-config</code>	is a script to get information about the installed version of Redland
<code>redland-db-upgrade</code>	upgrades older Redland databases to 0.9.12 format

sg3_utils-1.48

Introduction to sg3_utils

The sg3_utils package contains low level utilities for devices that use a SCSI command set. Apart from SCSI parallel interface (SPI) devices, the SCSI command set is used by ATAPI devices (CD/DVDs and tapes), USB mass storage devices, Fibre Channel disks, IEEE 1394 storage devices (that use the "SBP" protocol), SAS, iSCSI and FCoE devices (amongst others).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://sg.danny.cz/sg/p/sg3_utils-1.48.tar.xz
- Download MD5 sum: 0024393d2d2942cc081ce613d98db68a
- Download size: 1.2 MB
- Estimated disk space required: 27 MB
- Estimated build time: 0.2 SBU

Installation of sg3_utils

Install sg3_utils by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: rescan-scsi-bus.sh, scsi_logging_level, scsi_mandat, scsi_readcap, scsi_ready, scsi_satl, scsi_start, scsi_stop, scsi_temperature, sg_bg_ctl, sg_compare_and_write, sg_copy_results, sg_dd, sg_decode_sense, sg_emc_trespass, sg_format, sg_get_config, sg_get_lba_status, sg_ident, sg_inq,

sg_logs, sg_luns, sg_map, sg_map26, sg_modes, sg_opcodes, sg_persist, sg_prevent, sg_raw, sg_rbuf, sg_rdac, sg_read, sg_read_attr, sg_read_block_limits, sg_read_buffer, sg_read_long, sg_readcap, sg_reassign, sg_referrals, sg_rep_pip, sg_rep_zones, sg_requests, sg_reset, sg_reset_wp, sg_rmsn, sg_rtpg, sg_safte, sg_sanitize, sg_sat_identify, sg_sat_phys_event, sg_sat_read_gplog, sg_sat_set_features, sg_scan, sg_seek, sg_senddiag, sg_ses, sg_ses_microcode, sg_start, sg_stpg, sg_stream_ctl, sg_sync, sg_test_rwbuf, sg_timestamp, sg_turs, sg_unmap, sg_verify, sg_vpd, sg_wr_mode, sg_write_buffer, sg_write_long, sg_write_same, sg_write_verify, sg_write_x, sg_xcopy, sg_zone, sginfo, sgm_dd, and sgp_dd

Installed Library: libsgutils2.so

Installed Directories: None

Short Descriptions

rescan-scsi-bus.sh	adds or removes SCSI devices without having to reboot
scsi_logging_level	accesses Linux SCSI logging level information
scsi_mandat	checks SCSI device support for mandatory commands
scsi_readcap	does the SCSI READ CAPACITY command on disks
scsi_ready	does the SCSI TEST UNIT READY on devices
scsi_satl	checks for SCSI to ATA Translation (SAT) device support
scsi_start	starts one or more SCSI disks
scsi_stop	stops one or more SCSI disks
scsi_temperature	fetches the temperature of a SCSI device
sg_bg_ctl	performs a SCSI BACKGROUND CONTROL command on a device to perform "advanced background operations"
sg_compare_and_write	sends the SCSI COMPARE AND WRITE command to device
sg_copy_results	sends the SCSI RECEIVE COPY RESULTS command (XCOPY related)
sg_dd	copies data to and from files and devices. Specialised for devices that understand the SCSI command set
sg_decode_sense	takes SCSI sense data in binary or as a sequence of ASCII hexadecimal bytes and decodes it
sg_emc_trespass	changes ownership of a LUN from another Service-Processor to this one
sg_format	formats or resizes a SCSI disk (perhaps changes its block size)
sg_get_config	sends a SCSI GET CONFIGURATION command (MMC-4 +)
sg_get_elem_status	sends a SCSI GET PHYSICAL ELEMENT STATUS command to a device
sg_get_lba_status	sends the SCSI GET LBA STATUS command
sg_ident	sends a SCSI REPORT or SET IDENTIFYING INFORMATION command
sginfo	access mode page information for a SCSI (or ATAPI) device
sg_inq	sends a SCSI INQUIRY or ATA IDENTIFY (PACKET) DEVICE command and outputs the response
sg_logs	access log pages with SCSI LOG SENSE command
sg_luns	sends the SCSI REPORT LUNS command
sg_map	displays mapping between linux sg and other SCSI devices
sg_map26	maps a special file to a SCSI generic (sg) device (or vice versa)
sgm_dd	copies data to and from files and devices. Specialized for devices that understand the SCSI command set and does memory mapped transfers from sg devices
sg_modes	reads mode pages with the SCSI MODE SENSE command
sg_opcodes	reports information on supported SCSI commands or task management functions
sgp_dd	copies data to and from files and devices. Specialized for devices that understand the SCSI command set
sg_persist	sends a SCSI PERSISTENT RESERVE (IN or OUT) command to manipulate registrations and reservations
sg_prevent	sends a SCSI PREVENT ALLOW MEDIUM REMOVAL command
sg_raw	sends an arbitrary SCSI command to a device
sg_rbuf	reads data using the SCSI READ BUFFER command
sg_rdac	displays or modifies the RDAC Redundant Controller Page
sg_read	reads blocks of data continually from the same offset
sg_read_attr	performs a SCSI READ ATTRIBUTE command on a device
sg_read_block_limits	sends a SCSI READ BLOCK LIMITS command
sg_read_buffer	sends a SCSI READ BUFFER command

sg_readcap	sends a SCSI READ CAPACITY command
sg_read_long	sends a SCSI READ LONG command
sg_reassign	sends a SCSI REASSIGN BLOCKS command
sg_referrals	sends a SCSI REPORT REFERRALS command
sg_rep_pip	sends a SCSI REPORT PROVISIONING INITIALIZATION PATTERN command
sg_rep_zones	sends a SCSI REPORT ZONES command
sg_requests	sends one or more SCSI REQUEST SENSE commands
sg_reset	sends a SCSI device, target, bus or host reset; or checks reset state
sg_reset_wp	sends a SCSI RESET WRITE POINTER command
sg_rmsn	sends a SCSI READ MEDIA SERIAL NUMBER command
sg_rtpg	sends a SCSI REPORT TARGET PORT GROUPS command
sg_safte	fetches status information from a SCSI Accessed Fault-Tolerant Enclosure (SAF-TE) device
sg_sanitize	sends a SCSI SANITIZE command
sg_sat_identify	sends an ATA IDENTIFY (PACKET) DEVICE command via a SCSI to ATA Translation (SAT) layer
sg_sat_phys_event	sends an ATA READ LOG EXT command via a SAT pass through to fetch log page 11h which contains SATA phy event counters
sg_sat_read_gplog	sends an ATA READ LOG EXT command via a SCSI to ATA Translation (SAT) layer
sg_sat_set_features	sends an ATA SET FEATURES command via a SCSI to ATA Translation (SAT) layer
sg_scan	does a scan of sg devices (or given SCSI/ATAPI/ATA devices) and prints the results
sg_seek	performs a SCSI SEEK or PRE-FETCH command on a device and its cache
sg_senddiag	performs a SCSI SEND DIAGNOSTIC command
sg_ses	sends controls and fetches the current status from a SCSI Enclosure Services (SES) device
sg_ses_microcode	sends microcode to a SCSI enclosure
sg_start	sends a SCSI START STOP UNIT command to start, stop, load or eject medium
sg_stpg	sends a SCSI SET TARGET PORT GROUPS command
sg_stream_ctrl	performs a SCSI STREAM CONTROL or GET STREAM STATUS command on a device to open or close an I/O stream
sg_sync	sends a SCSI command to synchronize the cache
sg_test_rwbuf	tests the SCSI host adapter by issuing write and read operations on a device's buffer and calculating checksums
sg_timestamp	reports or sets the timestamp on a SCSI device
sg_turs	sends one or more SCSI TEST UNIT READY commands
sg_unmap	sends a SCSI UNMAP command
sg_verify	invokes SCSI VERIFY command(s) on a block device
sg_vpd	fetches Vital Product Data (VPD) pages using a SCSI INQUIRY command
sg_write_buffer	sends a SCSI WRITE BUFFER command
sg_write_long	sends a SCSI WRITE LONG command
sg_write_same	sends a SCSI WRITE SAME command
sg_write_verify	sends a SCSI WRITE AND VERIFY command
sg_write_x	performs SCSI WRITE commands on a device
sg_wr_mode	writes mode pages
sg_xcopy	copies data to and from files and devices using SCSI EXTENDED COPY (XCOPY)
sg_zone	performs SCSI ZONE commands on a device, such as OPEN, CLOSE, FINISH or SEQUENTIALIZE
libsutils2.so	contains the sg3_utils API functions

sysmon-qt-2.0.1

Introduction to sysmon-qt

The sysmon-qt package provides a program to monitor system values including CPU usage, memory usage, and system temperatures in a compact screen window.

This package is known to build and work properly using an LFS 12.2 platform.

Note

There is also a version of this package, sysmon-qt-1.1, at the same location as below that is based on [qt5-components-5.15.14](#). The build instructions are identical to these. The functionality is the same and it does not make sense to build both versions.

Package Information

- Download (HTTP): <https://github.com/lfs-book/sysmon-qt/releases/download/v2.0.1/sysmon-qt-2.0.1.tar.xz>
- Download MD5 sum: d1024657f90ac406cbf982e7189a200b
- Download size: 76 KB
- Estimated disk space required: 1.3 MB
- Estimated build time: 0.5 SBU

sysmon-qt Dependencies

Required

[Qt-6.7.2](#)

Recommended

[lm-sensors-3-6-0](#) (runtime)

Installation of sysmon-qt

Install sysmon-qt by running the following commands:

```
cd src          &&  
qmake6 sysmon-qt.pro &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: sysmon-qt

Installed Library: None

Installed Directories: None

Sysstat-12.7.6

Introduction to Sysstat

The Sysstat package contains utilities to monitor system performance and usage activity. Sysstat contains the `sar` utility, common to many commercial Unixes, and tools you can schedule via cron to collect and historize performance and activity data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sysstat.github.io/sysstat-packages/sysstat-12.7.6.tar.xz>
- Download MD5 sum: 16fd81728fec899f22155fdfcebece97
- Download size: 900 KB
- Estimated disk space required: 29 MB
- Estimated build time: 0.1 SBU

Sysstat Dependencies

There are no build-time requirements for this package; however, it is designed to be controlled by a cron daemon such as [Cron-3.2.1](#).

Installation of Sysstat

Install Sysstat by running the following commands:

```
sa_lib_dir=/usr/lib/sa      \
sa_dir=/var/log/sa          \
conf_dir=/etc/sysstat       \
./configure --prefix=/usr \
                  --disable-file-attr &&
make
```

This package does not come with a working test suite.

Now, as the `root` user:

```
make install
```

Install the systemd unit by running the following commands as the `root` user:

```
install -v -m644 sysstat.service /usr/lib/systemd/system/sysstat.service &&
install -v -m644 cron/sysstat-collect.service /usr/lib/systemd/system/sysstat-collect.service &&
install -v -m644 cron/sysstat-collect.timer /usr/lib/systemd/system/sysstat-collect.timer &&
install -v -m644 cron/sysstat-rotate.service /usr/lib/systemd/system/sysstat-rotate.service &&
install -v -m644 cron/sysstat-rotate.timer /usr/lib/systemd/system/sysstat-rotate.timer &&
install -v -m644 cron/sysstat-summary.service /usr/lib/systemd/system/sysstat-summary.service &&
install -v -m644 cron/sysstat-summary.timer /usr/lib/systemd/system/sysstat-summary.timer
```

Fix the systemd unit by running the following command as the `root` user:

```
sed -i "/^Also=/d" /usr/lib/systemd/system/sysstat.service
```

Command Explanations

`sa_lib_dir`: This environment variable specifies the location of the package-specific library directory.

`sa_dir`: This environment variable specifies the location of the directory containing the data files.

`conf_dir`: This environment variable specifies the location of the system configuration directory.

`--disable-file-attr`: Do not set attributes on files being installed. This parameter causes the installation to ignore the `man` group variable resulting in the man files having `root: root` ownership.

Note

Run `./configure --help` to see other influential environment variables you can pass to `configure`. You may want to use the `history` and `compressafter` variables to customize the amount of data files kept on the system.

Configuring Sysstat

Config Files

`/etc/sysconfig/sysstat` and `/etc/sysconfig/sysstat.ioconf`

Cron Information

To begin gathering Sysstat history information, you must add to, or create a privileged user's crontab. The history data location is `/var/log/sa`. The user running Sysstat utilities via cron must have write access to this location.

Below is an example of what to install in the crontab. Adjust the parameters to suit your needs. Use `man sa1` and `man sa2` for information about the commands.

```

# 8am-7pm activity reports every 10 minutes during weekdays
0 8-18 * * 1-5 /usr/lib/sa/sa1 600 6 &

# 7pm-8am activity reports every hour during weekdays
0 19-7 * * 1-5 /usr/lib/sa/sa1 &

# Activity reports every hour on Saturday and Sunday
0 * * * 0,6 /usr/lib/sa/sa1 &

# Daily summary prepared at 19:05
5 19 * * * /usr/lib/sa/sa2 -A &

```

Ensure you submit the revised crontab to the cron daemon.

System Startup Information

At system startup, a LINUX RESTART message must be inserted in the daily data file to reinitialize the kernel counters. This can be automated by enabling the previously installed systemd unit by running the `root` user:

```
systemctl enable sysstat
```

Contents

Installed Programs: cifsistat, iostat, mpstat, pidstat, sadf, sar, and tapestat

Installed Libraries: None

Installed Directories: /usr/lib/sa, /usr/share/doc/sysstat-12.7.6 and /var/log/sa

Short Descriptions

<code>cifsistat</code>	displays statistics about read and write operations on CIFS filesystems
<code>iostat</code>	reports CPU statistics and input/output statistics for devices and partitions
<code>mpstat</code>	writes activities for each available processor
<code>pidstat</code>	is used for monitoring individual tasks currently being managed by the Linux kernel
<code>sadf</code>	is used for displaying the contents of data files created by the <code>sar</code> command. But unlike <code>sar</code> , <code>sadf</code> can write its data in many different formats
<code>sar</code>	is used for displaying the contents of elected cumulative activity counters in the operating system
<code>tapestat</code>	is used for monitoring the activity of tape drives connected to a system

Systemd-256.4

Introduction to systemd

While systemd was installed when building LFS, there are many features provided by the package that were not included in the initial installation because Linux-PAM was not yet installed. The systemd package needs to be rebuilt to provide a working `systemd-logind` service, which provides many additional features for dependent packages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/systemd/systemd/archive/v256.4/systemd-256.4.tar.gz>
- Download MD5 sum: 03bd1ff158ec0bc55428c77a8f8495bd
- Download size: 15 MB
- Estimated disk space required: 328 MB (with tests)
- Estimated build time: 1.5 SBU (with tests using 4 cores)

systemd Dependencies

Recommended

Note

[Linux-PAM-1.6.1](#) is not strictly required to build systemd, but the main reason to rebuild systemd in BLFS (it's already built in LFS anyway) is for the `systemd-logind` daemon and the `pam_systemd.so` PAM module. [Linux-PAM-1.6.1](#) is required for them. All packages in BLFS book with a dependency on systemd expects it has been rebuilt with [Linux-PAM-1.6.1](#).

[Linux-PAM-1.6.1](#) and [Polkit-125](#) (runtime)

Optional

`btrfs-progs-6.10.1`, `cURL-8.9.1`, `cryptsetup-2.7.4`, `git-2.46.0`, `GnuTLS-3.8.7.1`, `iptables-1.8.10`, `libgcrypt-1.11.0`, `libidn2-2.3.7`, `libpwquality-1.4.5`, `libseccomp-2.5.5`, `libxkbcommon-1.7.0`, `make-ca-1.14`, `p11-kit-0.25.5`, `pcre2-10.44`, `qemu-9.0.2`, `gencode-4.1.1`, `rsync-3.3.0`, `sphinx-8.0.2`, `Valgrind-3.23.0`, `zsh-5.9` (for the zsh completions), `AppArmor`, `audit-userspace`, `bash-completion`, `jekyll`, `kexec-tools`, `libbpf`, `libdw`, `libfido2`, `libmicrohttpd`, `pefile`, `pyelftools`, `quota-tools`, `rpm`, `SELinux`, `systemtap`, `tpm2-tss` and `Xen`

Optional (to rebuild the manual pages)

`docbook-xml-4.5`, `docbook-xsl-nons-1.79.2`, `libxslt-1.1.42`, and `lxml-5.3.0` (to build the index of systemd manual pages)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/Logind>

Installation of systemd

Remove two unneeded groups, `render` and `sgx`, from the default udev rules:

```
sed -i -e 's/GROUP="render"/GROUP="video"/' \
-e 's/GROUP="sgx" // rules.d/50-udev-default.rules.in'
```

Rebuild systemd by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
--prefix=/usr \
--buildtype=release \
-D default-dnssec=no \
-D firstboot=false \
-D install-tests=false \
-D ldconfig=false \
-D man=auto \
-D sysusers=false \
-D rpmmacrosdir=no \
-D homed=disabled \
-D userdb=false \
-D mode=release \
-D pam=enabled \
-D pamconfdir=/etc/pam.d \
-D dev-kvm-mode=0660 \
-D nobody-group=nogroup \
-D sysupdate=disabled \
-D ukify=disabled \
-D docdir=/usr/share/doc/systemd-256.4 &&

ninja
```

Note

For the best test results, make sure you run the test suite from a system that is booted by the same systemd version you are rebuilding.

To test the results, issue: `ninja test`. The test named `test-stat-util` is known to fail if some kernel features are not enabled. If the test suite is run as the `root` user, some other tests may fail because they depend on various kernel configuration options.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D pamconfdir=/etc/pam.d`: Forces the PAM files to be installed in `/etc/pam.d` rather than `/usr/lib/pam.d`.

`-D userdb=false`: Removes a daemon that does not offer any use under a BLFS configuration. If you wish to enable the `userdbd` daemon, replace "false" with "true" in the above meson command.

`-D homed=disabled`: Removes a daemon that does not offer any use under a traditional BLFS configuration, especially using accounts created with `useradd`. To enable `systemd-homed`, first ensure that you have [cryptsetup-2.7.4](#) and [libpwquality-1.4.5](#) installed, and then change "disabled" to "enabled" in the above `meson setup` command.

`-D ukify=disabled`: Removes a script for combining a kernel, an initramfs, and a kernel command line etc. into an UEFI application which can be loaded by the UEFI firmware to start the embedded Linux kernel. It's not needed for booting a BLFS system with UEFI if following [Using GRUB to Set Up the Boot Process with UEFI](#). And, it requires the `pefile` Python module at runtime, so if it's enabled but `pefile` is not installed, in the test suite one test for it will fail. To enable `systemd-ukify`, install the `pefile` module and then change "disabled" to "enabled" in the above `meson setup` command.

Configuring systemd

The `/etc/pam.d/system-session` file needs to be modified and a new file needs to be created in order for `systemd-logind` to work correctly. Run the following commands as the `root` user:

```
grep 'pam_systemd' /etc/pam.d/system-session ||
cat >> /etc/pam.d/system-session << "EOF"
# Begin Systemd addition

session required pam_loginuid.so
session optional pam_systemd.so

# End Systemd addition
EOF

cat > /etc/pam.d/systemd-user << "EOF"
# Begin /etc/pam.d/systemd-user

account required pam_access.so
account include system-account

session required pam_env.so
session required pam_limits.so
session required pam_loginuid.so
session optional pam_keyinit.so force revoke
session optional pam_systemd.so

auth required pam_deny.so
password required pam_deny.so

# End /etc/pam.d/systemd-user
EOF
```

As the `root` user, replace the running `systemd` manager (the `init` process) with the `systemd` executable newly built and installed:

```
systemctl daemon-reexec
```

Important

Now ensure [Shadow-4.16.0](#) has been already rebuilt with [Linux-PAM-1.6.1](#) support first, then logout, and login again. This ensures the running login session registered with `systemd-logind` and a per-user `systemd` instance running for each user owning a login session. Many BLFS packages listing `Systemd` as a dependency needs the `systemd-logind` integration and/or a running per-user `systemd` instance.

Warning

If upgrading from a previous version of systemd and an initrd is used for system boot, you should generate a new initrd before rebooting the system.

Contents

A list of the installed files, along with their short descriptions can be found at [..../..../..../lfs/view/12.2-systemd/chapter08/systemd.html#contents-systemd](#).

Listed below are the newly installed programs along with short descriptions.

Installed Programs: `homectl` (optional), `systemd-cryptenroll` (if `cryptsetup-2.7.4` is installed), and `userdbctl` (optional)

Short Descriptions

<code>homectl</code>	is a tool to create, remove, change, or inspect a home directory managed by <code>systemd-homed</code> ; note that it's useless for the classic UNIX users and home directories which we are using in LFS/BLFS book
<code>systemd-cryptenroll</code>	Is used to enroll or remove a system from full disk encryption, as well as set and query private keys and recovery keys
<code>userdbctl</code>	inspects users, groups, and group memberships
<code>pam_systemd.so</code>	is a PAM module used to register user sessions with the <code>systemd</code> login manager, <code>systemd-logind</code>

UDisks-2.10.1

Introduction to UDisks

The UDisks package provides a daemon, tools and libraries to access and manipulate disks and storage devices.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/storaged-project/udisks/releases/download/udisks-2.10.1/udisks-2.10.1.tar.bz2>
- Download MD5 sum: 613af9bfea52cde74d2ac34d96de544d
- Download size: 1.8 MB
- Estimated disk space required: 44 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

UDisks Dependencies

Required

[libatasmart-0.19](#), [libblockdev-3.1.1](#), [libgudev-238](#), [libxslt-1.1.42](#), and [Polkit-125](#)

Recommended

[Systemd-256.4](#) (runtime)

Optional (Required if building GNOME)

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[D-Bus Python-1.3.2](#) (for the integration tests), [GTK-Doc-1.34.0](#), [LVM2-2.03.26](#), [PyGObject-3.48.2](#) (for the integration tests), [exFAT](#), and [libiscsi](#)

Optional Runtime Dependencies

[btrfs-progs-6.10.1](#), [dosfstools-4.2](#), [gptfdisk-1.0.10](#), [mdadm-4.3](#), and [xfsprogs-6.9.0](#)

Installation of UDisks

Install UDisks by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --localstatedir=/var \
            --disable-static   \
            --enable-available-modules &&
make
```

To test the results, issue: `make check`. A more thorough test can be run with `make ci`. You must first create the directories `/var/run/udisks2` and `/var/lib/udisks2`, and the optional python modules should be present.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

`--enable-available-modules`: This switch enables additional UDisks2 functionalities if [libblockdev-3.1.1](#) has been built with optional dependencies.

Contents

Installed Programs: `udisksctl` and `umount.udisks2`

Installed Library: `libudisks2.so`

Installed Directories: `/etc/udisks2`, `/usr/include/udisks2`, `/usr/libexec/udisks2`, `/usr/share/gtk-doc/html/udisks2`, and `/var/lib/udisks2`

Short Descriptions

- | | |
|-----------------------------|--|
| <code>udisksctl</code> | is a command-line program used to interact with the <code>udisksd</code> daemon |
| <code>umount.udisks2</code> | is a command-line program used to unmount file systems that have been mounted by the UDisks daemon |
| <code>libudisks2.so</code> | contains the UDisks API functions |

UnRar-7.0.9

Introduction to UnRar

The UnRar package contains a `RAR` extraction utility used for extracting files from `RAR` archives. `RAR` archives are usually created with WinRAR, primarily in a Windows environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.rarlab.com/rar/unrarsrc-7.0.9.tar.gz>
- Download MD5 sum: a18509f604f31cf7a73473ee97d08ce5
- Download size: 256 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: 0.2 SBU

Installation of UnRar

Note

This package extracts the tarball to the unversioned directory `unrar` and not the expected directory `unrar-7.0.9`.

Install UnRar by running the following commands:

```
make -f makefile
```

This package does not come with a test suite.

Now, as the `root` user:

```
install -v -m755 unrar /usr/bin
```

Contents

Installed Program: unrar

Installed Libraries: None

Installed Directories: None

Short Descriptions

`unrar` uncompressed a `RAR` archive

UnZip-6.0

Introduction to UnZip

The UnZip package contains `ZIP` extraction utilities. These are useful for extracting files from `ZIP` archives. `ZIP` archives are created with PKZIP or Info-ZIP utilities, primarily in a DOS environment.

This package is known to build and work properly using an LFS 12.2 platform.

Caution

The previous version of the UnZip package had some locale related issues. Currently there are no BLFS editors capable of testing these locale issues. Therefore, the locale related information is left on this page, but has not been tested. A more general discussion of these problems can be found in the [Program Assumes Encoding](#) section of the [Locale Related Issues](#) page.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/infozip/unzip60.tar.gz>
- Download MD5 sum: 62b490407489521db863b523a7f86375
- Download size: 1.3 MB
- Estimated disk space required: 9 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/unzip-6.0-consolidated_fixes-1.patch
- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/unzip-6.0-gcc14-1.patch>

UnZip Locale Issues

Note

Use of UnZip in the JDK, Mozilla, DocBook or any other BLFS package installation is not a problem, as BLFS instructions never use UnZip to extract a file with non-ASCII characters in the file's name.

These issues are thought to be fixed in the patch. But since none of the editors have data to test this, the following workarounds are retained in case they might still be needed.

The UnZip package assumes that filenames stored in the ZIP archives created on non-Unix systems are encoded in CP850, and that they should be converted to ISO-8859-1 when writing files onto the filesystem. Such assumptions are not always

valid. In fact, inside the ZIP archive, filenames are encoded in the DOS codepage that is in use in the relevant country, and the filenames on disk should be in the locale encoding. In MS Windows, the `OemToChar()` C function (from `User32.DLL`) does the correct conversion (which is indeed the conversion from CP850 to a superset of ISO-8859-1 if MS Windows is set up to use the US English language), but there is no equivalent in Linux.

When using `unzip` to unpack a ZIP archive containing non-ASCII filenames, the filenames are damaged because `unzip` uses improper conversion when any of its encoding assumptions are incorrect. For example, in the ru_RU.KOI8-R locale, conversion of filenames from CP866 to KOI8-R is required, but conversion from CP850 to ISO-8859-1 is done, which produces filenames consisting of undecipherable characters instead of words (the closest equivalent understandable example for English-only users is rot13). There are several ways around this limitation:

1) For unpacking ZIP archives with filenames containing non-ASCII characters, use [WinZip](#) while running the [Wine](#) Windows emulator.

2) Use `bsdtar -xf` from [libarchive-3.7.4](#) to unpack the ZIP archive. Then fix the damage made to the filenames using the `convmv` tool (<https://j3e.de/linux/convmv/>). The following is an example for the zh_CN.UTF-8 locale:

```
convmv -f cp936 -t utf-8 -r --nosmart --notest \
</path/to/unzipped/files>
```

Installation of UnZip

First apply the patches:

```
patch -Np1 -i .../unzip-6.0-consolidated_fixes-1.patch
patch -Np1 -i .../unzip-6.0-gcc14-1.patch
```

Now compile the package:

```
make -f unix/Makefile generic
```

The test suite does not work for target `generic`.

Now, as the `root` user:

```
make prefix=/usr MANDIR=/usr/share/man/man1 \
-f unix/Makefile install
```

Command Explanations

`make -f unix/Makefile generic`: This target begins by running a configure script (unlike the older targets such as `linux` and `linux_noasm`) which creates a flags file that is then used in the build. This ensures that the 32-bit x86 build receives the right flags to unzip files which are larger than 2 GB when extracted.

Contents

Installed Programs: `funzip`, `unzip`, `unzipfsx`, `zipgrep`, and `zipinfo`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>funzip</code>	allows the output of <code>unzip</code> commands to be redirected
<code>unzip</code>	lists, tests or extracts files from a ZIP archive
<code>unzipfsx</code>	is a self-extracting stub that can be prepended to a ZIP archive. Files in this format allow the recipient to decompress the archive without installing UnZip
<code>zipgrep</code>	searches files in a ZIP archive for lines matching a pattern
<code>zipinfo</code>	produces technical information about the files in a ZIP archive, including file access permissions, encryption status, type of compression, etc

UPower-1.90.4

Introduction to UPower

The UPower package provides an interface for enumerating power devices, listening to device events and querying history and statistics. Any application or service on the system can access the `org.freedesktop.UPower` service via the system

message bus.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/upower/upower/-/archive/v1.90.4/upower-v1.90.4.tar.bz2>
- Download MD5 sum: 2399845217a9c6e4510f66033fc7cf6
- Download size: 140 KB
- Estimated disk space required: 5.5 MB (add 2.6 MB for tests)
- Estimated build time: less than 0.1 SBU (add 0.6 SBU for tests)

UPower Dependencies

Required

[libgudev-238](#) and [libusb-1.0.27](#)

Optional (Required if building GNOME)

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#), [libxslt-1.1.42](#), [docbook-xsl-nons-1.79.2](#), [PyGObject-3.48.2](#), [dbusmock-0.32.1](#), [umockdev-0.18.3](#) (for part of the test suite), and [libimobiledevice](#)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
General setup --->
  *-- Namespaces support --->
    [*] User namespace
[NAMESPACES]
[USER_NS]
```

Installation of UPower

Install UPower by running the following commands:

```
mkdir build
cd build
meson setup ..
  --prefix=/usr
  --buildtype=release
  -D gtk-doc=false
  -D man=false
ninja
```

To test the results, issue: `lc_ALL=C ninja test`. The test suite should be run from a local GUI session started with `dbus-launch`. On 32-bit machines, one test will fail due to rounding errors: `Tests.test_battery_energy_charge_mixed`. On some systems, two tests relating to the headphone hotplug feature are known to fail. Those can be safely ignored since the functionality still works.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D gtk-doc=false`: Prevents building the documentation. Remove this if you have GTK-Doc installed and wish to build the documentation.

`-D man=false`: Prevents building the manual pages. Remove this if you have [libxslt-1.1.42](#) and [docbook-xsl-nons-1.79.2](#) installed and wish to build the manual pages.

Contents

Installed Program: upower
Installed Libraries: libupower-glib.so
Installed Directories: /etc/UPower, /usr/include/libupower-glib, and /var/lib/upower

Short Descriptions

`upower` is the UPower command line tool
`libupower-glib.so` contains the UPower API functions

usbutils-017

Introduction to USB Utils

The USB Utils package contains utilities used to display information about USB buses in the system and the devices connected to them.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://kernel.org/pub/linux/utils/usb/usbutils-017.tar.xz>
- Download MD5 sum: 8ff21441faf2e8128e4810b3d6e49059
- Download size: 168 KB
- Estimated disk space required: 4.4 MB
- Estimated build time: 0.1 SBU

USB Utils Dependencies

Required

[libusb-1.0.27](#)

Recommended

[hwdata-0.385](#) (runtime)

Installation of USB Utils

Install USB Utils by running the following commands:

```
./configure --prefix=/usr --datadir=/usr/share/hwdata &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

For the `usb.ids` data file, install the [hwdata-0.385](#) package.

The script `lsusb.py` displays information in a more easily readable form than `lsusb`. To find the options, use `lsusb.py -h`. One form of use recommended by the developer is `lsusb.py -ciu`.

Contents

Installed Programs: lsusb, lsusb.py, usb-devices, and usbhid-dump
Installed Libraries: None
Installed Directories: None

Short Descriptions

`lsusb` is a utility for displaying information about all USB buses in the system and all devices connected to them, but not in human friendly form

<code>lsusb.py</code>	displays information about all USB buses in the system and all devices connected to them in reasonable human friendly form
<code>usb-devices</code>	is a shell script that displays details of USB buses and devices connected to them. It is designed to be used if /proc/bus/usb/devices is not available on your system
<code>usbhid-dump</code>	is used to dump report descriptors and streams from HID (human interface device) interfaces of USB devices

Which-2.21 and Alternatives

The presence or absence of the `which` program in the main LFS book is probably one of the most contentious issues on the mailing lists. It has resulted in at least one flame war in the past. To hopefully put an end to this once and for all, presented here are two options for equipping your system with `which`. The question of which “`which`” is for you to decide.

Introduction to GNU Which

The first option is to install the actual GNU `which` package.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/which/which-2.21.tar.gz>
- Download MD5 sum: 097ff1a324ae02e0a3b0369f07a7544a
- Download size: 148 KB
- Estimated disk space required: 1 MB
- Estimated build time: less than 0.1 SBU

Installation of Which

Install `which` by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: `which`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`which` shows the full path of (shell) commands installed in your `PATH`

The 'which' Script

The second option (for those who don't want to install the package) is to create a simple script (execute as the `root` user):

```
cat > /usr/bin/which << "EOF"
#!/bin/bash
type -pa "$@" | head -n 1 ; exit ${PIPESTATUS[0]}
EOF
chmod -v 755 /usr/bin/which
chown -v root:root /usr/bin/which
```

This should work OK and is probably the easiest solution for most cases, but is not the most comprehensive implementation.

xdotool-3.20211022.1

Introduction to xdotool

The xdotool package provides the capabilities to simulate keyboard input and mouse activity, move and resize windows, etc. It does this using X11's XTEST extension and other Xlib functions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/jordansissel/xdotool/releases/download/v3.20211022.1/xdotool-3.20211022.1.tar.gz>
- Download MD5 sum: 9fd993a251a7c38b32381503544b0dd7
- Download size: 116 KB
- Estimated disk space required: 2.1 MB
- Estimated build time: less than 0.1 SBU

Installation of xdotool

Install xdotool by running the following commands:

```
make WITHOUT_RPATH_FIX=1
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr INSTALLMAN=/usr/share/man install
```

Contents

Installed Program: xdotool

Installed Libraries: libxdo.so

Installed Directories: None

Zip-3.0

Introduction to Zip

The Zip package contains Zip utilities. These are useful for compressing files into `.ZIP` archives.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/infozip/zip30.tar.gz>
- Download MD5 sum: 7b74551e63f8ee6aab6fbc86676c0d37
- Download size: 1.1 MB
- Estimated disk space required: 6.4 MB
- Estimated build time: 0.1 SBU

Installation of Zip

Install Zip by running the following commands:

```
make -f unix/Makefile generic CC="gcc -std=gnu89"
```

This package does not come with a test suite.

Now, as the `root` user:

```
make prefix=/usr MANDIR=/usr/share/man/man1 -f unix/Makefile install
```

Command Explanations

`CC="gcc -std=gnu89"`: This parameter overrides the `cc` variable that is set to `cc` in the `unix/Makefile` file. On LFS `cc` is a symlink to `gcc`, and it uses `-std=gnu17` (ISO 9899:2017 with GNU extensions) as the default but Zip is a "legacy" package where some grammar constructs are invalid in ISO 9899:1999 and newer standards.

`prefix=/usr`: This parameter overrides the `prefix` variable that is set to `/usr/local` in the `unix/Makefile` file.

Contents

Installed Programs: zip, zipcloak, zipnote, and zipsplit

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>zip</code>	compresses files into a <code>ZIP</code> archive
<code>zipcloak</code>	is a utility to encrypt and decrypt a <code>ZIP</code> archive
<code>zipnote</code>	reads or writes comments stored in a <code>ZIP</code> file
<code>zipsplit</code>	is a utility to split <code>ZIP</code> files into smaller files

Chapter 13. Programming

A base LFS system can be used as a development platform, however the base system only includes language support for C, C++, Perl, and Python. This chapter provides instructions to build many popular programming environments to greatly expand your system's development capabilities.

Cbindgen-0.27.0

Introduction to Cbindgen

Cbindgen can be used to generate C bindings for Rust code.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mozilla/cbindgen/archive/v0.27.0/cbindgen-0.27.0.tar.gz>
- Download MD5 sum: 9349f81198b82c970822c452d08f25e5
- Download size: 224 KB
- Estimated disk space required: 121 MB (add 600 MB for tests)
- Estimated build time: 0.5 SBU (add 0.4 SBU for tests), both on a 4-core machine

cbindgen Dependencies

Required

[rustc-1.80.1](#)

Note

An Internet connection is needed for building this package.

Installation of cbindgen

Install cbindgen by running the following commands:

```
cargo build --release
```

To test the results, issue: `cargo test --release`. Three tests in `profile.rs` are known to fail because they expect some Rust unstable features disabled in the BLFS [rustc-1.80.1](#) configuration.

Now, as the `root` user:

```
install -Dm755 target/release/cbindgen /usr/bin/
```

Contents

Installed Program: cbindgen

Installed Library: none

Installed Directory: none

Short Descriptions

`cbindgen` generates C bindings for Rust code

Clisp-2.49

Introduction to Clisp

GNU Clisp is a Common Lisp implementation which includes an interpreter, compiler, debugger, and many extensions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/clisp/latest/clisp-2.49.tar.bz2>
- Download MD5 sum: 1962b99d5e530390ec3829236d168649
- Download size: 7.8 MB
- Estimated disk space required: 163 MB (add 8 MB for tests)
- Estimated build time: 0.9 SBU (1.2 SBU with tests)

Additional Downloads

- Optional patch: https://www.linuxfromscratch.org/patches/blfs/12.2/clisp-2.49-readline7_fixes-1.patch (required if building against libffcall)

Clisp Dependencies

Recommended

[libsigsegv-2.14](#)

Optional

[libnsl-2.0.1](#) and [libffcall](#)

Installation of Clisp

Note

This package does not support parallel build.

If you are building on a 32-bit system, work around a bug in GCC caused by the latest version of binutils:

```
case $(uname -m) in
    i?86) export CFLAGS="${CFLAGS:--O2 -g} -falign-functions=4" ;;
esac
```

Remove two tests which fail for unknown reasons:

```
sed -i -e '/socket/d' -e '"streams"/d' tests/tests.lisp
```

Install Clisp by running the following commands:

If you are building clisp against libffcall, apply the patch to fix a build failure with current readline:

```
patch -Np1 -i ../clisp-2.49-readline7_fixes-1.patch
```

Install Clisp by running the following commands:

```
mkdir build &&
cd      build &&

./configure --srcdir=../           \
            --prefix=/usr          \
            --docdir=/usr/share/doc/clisp-2.49 \
            --with-libsigsegv-prefix=/usr &&

ulimit -S -s 16384 &&
make -j1
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`ulimit -s 16384`: this increases the maximum stack size, as recommended by the `configure`.

`--docdir=/usr/share/doc/clisp-2.49`: this switch forces the HTML documentation to be installed into a versioned directory, rather than into `/usr/share/html/`.

`--with-libsigsegv-prefix=/usr`: use this to tell `configure` that you have installed `libsigsegv` in `/usr`, otherwise it will not be found.

`--with-libffcall-prefix=/usr`: use this to tell `configure` that you have installed the optional `libffcall` package in `/usr`, otherwise like `libsigsegv` it will not be found.

Contents

Installed Programs: `clisp` and `clisp-link`

Installed Libraries: various static libraries in `/usr/lib/clisp-2.49/base/`

Installed Directories: `/usr/lib/clisp-2.49`, `/usr/share/doc/clisp-2.49`, and `/usr/share/emacs/site-lisp`

Short Descriptions

`clisp` is an ANSI Common Lisp compiler, interpreter, and debugger

`clisp-link` is used to link an external module to `clisp`

CMake-3.30.2

Introduction to CMake

The CMake package contains a modern toolset used for generating Makefiles. It is a successor of the auto-generated `configure` script and aims to be platform- and compiler-independent. A significant user of CMake is KDE since version 4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cmake.org/files/v3.30/cmake-3.30.2.tar.gz>
- Download MD5 sum: 506cada2fe0c12dc6b3f282b8f72d325
- Download size: 11 MB

- Estimated disk space required: 452 MB (add 1.3 GB for tests)
- Estimated build time: 3.0 SBU (add 4.0 SBU for tests, both using parallelism=4)

CMake Dependencies

Recommended

[cURL-8.9.1](#), [libarchive-3.7.4](#), [libuv-1.48.0](#), and [nghttp2-1.62.1](#)

Optional

[GCC-14.2.0](#) (for gfortran), [git-2.46.0](#) (for use during tests), [Mercurial-6.8.1](#) (for use during tests), [OpenJDK-22.0.2](#) (for use during tests), [Qt-6.7.2](#) (for the Qt-based GUI), [Sphinx-8.0.2](#) (for building documents), [Subversion-1.14.3](#) (for testing), [cppdap](#), [jsoncpp](#), and [rhash](#)

Installation of CMake

Install CMake by running the following commands:

```
sed -i '/"/lib64"/s/64//' Modules/GNUInstallDirs.cmake &&
./bootstrap --prefix=/usr \
             --system-libs \
             --mandir=/share/man \
             --no-system-jsoncpp \
             --no-system-cppdap \
             --no-system-librhash \
             --docdir=/share/doc/cmake-3.30.2 &&
make
```

To test the results, issue: `LC_ALL=en_US.UTF-8 bin/ctest -j<N> -O cmake-3.30.2-test.log`, where `<N>` is an integer between 1 and the number of system cores. Setting `LC_ALL` is needed to prevent some test failures when some of the locale variables are set to non English locales.

If you want to investigate a problem with a given "problem1-test", use `bin/ctest -R "problem1-test"` and, to omit it, use `bin/ctest -E "problem1-test"`. These options can be used together: `bin/ctest -R "problem1-test" -E "problem2-test"`. Option `-N` can be used to display all available tests, and you can run `bin/ctest` for a sub-set of tests by using separated by spaces names or numbers as options. Option `--help` can be used to show all options.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ... Modules/GNUInstallDirs.cmake`: This command disables applications using `cmake` from attempting to install files in `/usr/lib64/`.

`--system-libs`: This switch forces the build system to link against the system installed version for all needed libraries but those explicitly specified via a `--no-system-*` option.

`--no-system-jsoncpp` and `--no-system-cppdap`: These switches remove the JSON-C++ library from the list of system libraries. A bundled version of that library is used instead.

`--no-system-librhash`: This switch removes the librhash library from the list of system libraries used. A bundled version of that library is used instead.

`--no-system-{curl,libarchive,libuv,nghttp2}`: Use the corresponding option in the list for the `bootstrap` if one of the recommended dependencies is not installed. A bundled version of the dependency will be used instead.

`--qt-gui`: This switch enables building the Qt-based GUI for CMake.

`--parallel=`: This switch enables performing the CMake bootstrap with multiple jobs at one time. It's not needed if the `MAKEFLAGS` variable has been already set for using multiple processors following [Using Multiple Processors](#).

Contents

Installed Programs: `ccmake`, `cmake`, `cmake-gui` (optional), `cpack`, and `ctest`

Installed Libraries: None

Installed Directories: /usr/share/cmake-3.30 and /usr/share/doc/cmake-3.30.2

Short Descriptions

<code>ccmake</code>	is a curses based interactive frontend to <code>cmake</code>
<code>cmake</code>	is the makefile generator
<code>cmake-gui</code>	(optional) is the Qt-based frontend to <code>cmake</code>
<code>cpack</code>	is the CMake packaging program
<code>ctest</code>	is a testing utility for cmake-generated build trees

Doxygen-1.12.0

Introduction to Doxygen

The Doxygen package contains a documentation system for C++, C, Java, Objective-C, Corba IDL and to some extent PHP, C# and D. It is useful for generating HTML documentation and/or an off-line reference manual from a set of documented source files. There is also support for generating output in RTF, PostScript, hyperlinked PDF, compressed HTML, and Unix man pages. The documentation is extracted directly from the sources, which makes it much easier to keep the documentation consistent with the source code.

You can also configure Doxygen to extract the code structure from undocumented source files. This is very useful to quickly find your way in large source distributions. Used along with Graphviz, you can also visualize the relations between the various elements by means of include dependency graphs, inheritance diagrams, and collaboration diagrams, which are all generated automatically.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://doxygen.nl/files/doxygen-1.12.0.src.tar.gz>
- Download MD5 sum: 3ab97fd76c6fe090946c836c5051182b
- Download size: 8.1 MB
- Estimated disk space required: 248 MB (with tests)
- Estimated build time: 1.9 SBU (with tests; both using parallelism=4)

Doxygen Dependencies

Required

[CMake-3.30.2](#) and [git-2.46.0](#)

Recommended

[Qt-6.7.2](#) (for doxywizard)

Optional

[Graphviz-12.1.0](#), [ghostscript-10.03.1](#), [libxml2-2.13.3](#) (required for the tests), [LLVM-18.1.7](#) (with clang), [qt5-components-5.15.14](#) (deprecated), [texlive-20240312](#) (or [install-tl-unx](#)), [xapian-1.4.26](#) (for doxyindexer), and [javacc](#)

Installation of Doxygen

First, fix up some python scripts:

```
grep -rl '^#!.*python$' | xargs sed -i '1s/python/\&3/'
```

Install Doxygen by running the following commands:

```
mkdir -v build &&
cd      build &&

cmake -G "Unix Makefiles" \
      -D CMAKE_BUILD_TYPE=Release \
      -D CMAKE_INSTALL_PREFIX=/usr \
      -D build_wizard=ON \
      -D force_qt=Qt6 \
```

```
-W no-dev ... &&  
make
```

To test the results, issue: `make tests`. One test, `012_cite.dox`, is known to fail if [texlive-20240312](#) or [install-tl-unx](#) is not installed.

If you wish to generate the package documentation, you must have Python, TeX Live (for HTML docs) and Ghostscript (for PDF docs) installed, then issue the following command:

```
cmake -D build_doc=ON \  
      -D DOC_INSTALL_DIR=share/doc/doxygen-1.12.0 \  
      .. &&  
make docs
```

Now, as the `root` user:

```
make install &&  
install -vm644 ../doc/*.* /usr/share/man/man1
```

If you have generated the package documentation, then the man pages are automatically installed, and you do not need to run the last `install ...` command.

Command Explanations

`-D build_wizard=OFF`: Use this switch if Qt6 is not installed.

`-D build_search=ON`: Use this switch if xapian is installed and you wish to build external search tools (`doxysearch.cgi` and `doxyindexer`).

`-D force_qt6=ON`: Use this switch to build `doxywizard` with Qt6 even if Qt5 is installed.

`-D use_libclang=ON`: Use this switch if llvmparser and clang are installed, to add support for libclang parsing.

Configuring Doxygen

There is no real configuration necessary for the Doxygen package although three additional packages are required if you wish to use extended capabilities. If you require formulas to create PDF documentation, then you must have [texlive-20240312](#) installed. If you require formulas to convert PostScript files to bitmaps, then you must have [ghostscript-10.03.1](#) installed.

Contents

Installed Programs: doxygen and optionally, doxywizard, doxyindexer and doxysearch.cgi

Installed Libraries: None

Installed Directory: /usr/share/doc/doxygen-1.12.0

Short Descriptions

<code>doxygen</code>	is a command-line based utility used to generate template configuration files and then generate documentation from these templates. Use <code>doxygen --help</code> for an explanation of the command-line parameters
<code>doxywizard</code>	is a GUI front-end for configuring and running <code>doxygen</code>
<code>doxyindexer</code>	generates a search index called <code>doxysearch.db</code> from one or more search data files produced by <code>doxygen</code> . See, e.g. https://javacc.github.io/javacc/
<code>doxysearch.cgi</code>	is a CGI program to search the data indexed by <code>doxyindexer</code>

dtc-1.7.1

Introduction to Dtc

The dtc package contains the Device Tree Compiler for working with device tree source and binary files and also libfdt, a utility library for reading and manipulating device trees in the binary format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://kernel.org/pub/software/utils/dtc/dtc-1.7.1.tar.xz>
- Download MD5 sum: 8dc1c58c84ca19f825f6939801b44c52
- Download size: 168 KB
- Estimated disk space required: 15 MB
- Estimated build time: 0.3 SBU

Dtc Dependencies

Optional

[libyaml-0.2.5](#), [SWIG-4.2.1](#), and [texlive-20240312](#)

Installation of Dtc

Install dtc by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr           \
            --buildtype=release   \
            -D python=disabled .. &&
ninja
```

To test the results, issue: `meson test -v`.

Now, as the `root` user:

```
ninja install
```

Still as the `root` user, remove the useless static library:

```
rm /usr/lib/libfdt.a
```

If you have [texlive-20240312](#) installed, you can build the PDF format of the documentation by issuing the following command:

```
pushd ./Documentation
latexmk -bibtex --pdf dtc-paper &&
latexmk -bibtex --pdf dtc-paper -c
popd
```

To install the documentation, as the `root` user issue the following command:

```
cp -R ./Documentation -T /usr/share/doc/dtc-1.7.1
```

If you have installed [SWIG-4.2.1](#) and you wish to install the Python 3 binding of this package, build the Python 3 module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir ..
```

As the `root` user, install the Python 3 module:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user libfdt
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D python=disabled`: This switch prevents building the Python 3 binding with the deprecated method (running `setup.py` directly). We will build the Python 3 binding with the `pip3 wheel` command separately if wanted.

Contents

Installed Programs: convert-dtsv0, dtc, dtdiff, fdtdump, fdtget, fdtoverlay, and fdtput

Installed Library: libfdt.so

Installed Directory: /usr/lib/python3.12/site-packages/libfdt-1.7.1.dist-info and /usr/share/doc/dtc-1.7.1 (optionally)

Short Descriptions

<code>convert-dtsv0</code>	converts device tree v0 source to device tree v1
<code>dtc</code>	compiles device tree source (dts) to device tree binary blob (dtb), or de-compiles dtb to dts
<code>dtdiff</code>	compares two different device tree
<code>fdtdump</code>	prints a readable version of a flat device-tree file
<code>fdtget</code>	reads values from device-tree
<code>fdtoverlay</code>	applies a number of overlays to a base device tree blob
<code>fdtput</code>	writes a property value to a device tree
<code>libfdt.so</code>	is a utility library for reading and manipulating device trees in the binary format

GCC-14.2.0

Introduction to GCC

The GCC package contains the GNU Compiler Collection. This page describes the installation of compilers for the following languages: C, C++, Fortran, Objective C, Objective C++, Go, and Modula2. Since C and C++ are installed in LFS, this page is either for upgrading C and C++, or for installing additional compilers.

Note

Additional languages, among which D and Ada, are available in the collection. D and Ada have a binary bootstrap requirement for the first installation, so their installation is not described here. To install them, you can proceed along the same lines as below after installing the corresponding compiler from a binary package, adding `ada` or `d` to the `--enable-languages` line.

This package is known to build and work properly using an LFS 12.2 platform.

Caution

If you are upgrading GCC from any other version prior to 14.2.0, then you must be careful compiling 3rd party kernel modules. You should ensure that the kernel and all its native modules are also compiled using the same version of GCC that you use to build the 3rd party module. This issue does not affect native kernel (and kernel modules) updates, as the instructions below are a complete reinstallation of GCC. If you have existing 3rd party modules installed, ensure they are recompiled using the updated version of GCC.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/gcc/gcc-14.2.0/gcc-14.2.0.tar.xz>
- Download MD5 sum: 2268420ba02dc01821960e274711bde0
- Download size: 88 MB
- Estimated disk space required: 11 GB (3.4 GB installed with all listed languages; add 1.6 GB for tests)
- Estimated build time: 14 SBU (add 34 SBU for tests; both with parallelism=8)

GCC Dependencies

Optional

[GDB-15.1](#), [Graphviz-12.1.0](#) (some tests use it if installed; note that if it's installed but not built with [libpng-1.6.43](#) these tests will fail), [Valgrind-3.23.0](#) (for tests), and [ISL](#) (to enable graphite optimization)

Installation of GCC

Important

Even if you specify only languages other than C and C++ to the `./configure` command below, the installation process will overwrite your existing GCC C and C++ compilers and libraries. Running the full suite of tests is recommended.

Do not continue with the `make install` command until you are confident the build was successful. You can compare your test results with those found at <https://gcc.gnu.org/ml/gcc-testresults/>. You may also want to refer to the information found in the GCC section of Chapter 8 in the LFS book ([../../../../lfs/view/12.2-systemd/chapter08/gcc.html](#)).

The instructions below are intentionally performing a “bootstrap” process. Bootstrapping is needed for robustness and is highly recommended when upgrading the compilers version. To disable bootstrap anyway, add `--disable-bootstrap` to the `./configure` options below.

Install GCC by running the following commands:

```
case $(uname -m) in
x86_64)
    sed -i.orig '/m64=/s/lib64/lib/' gcc/config/i386/t-linux64
;;
esac

mkdir build      &&
cd   build      &&
./configure      \
--prefix=/usr \
--disable-multilib \
--with-system-zlib \
--enable-default-pie \
--enable-default-ssp \
--enable-host-pie \
--disable-fixincludes \
--enable-languages=c,c++,fortran,go,objc,obj-c++,m2 &&
make
```

If running tests, as in LFS, remove/fix several known test failures:

```
sed -e '/cpython/d'           -i ..//gcc/testsuite/gcc.dg/plugin/plugin.exp
sed -e 's/no-pic /&-no-pie /' -i ..//gcc/testsuite/gcc.target/i386/pr113689-1.c
sed -e 's/300000/(1|300000)/' -i ..//libgomp/testsuite/libgomp.c-c++-common/pr109062.c
sed -e 's/{ target nonpic } //\' \
-e '/GOTPCREL/d'            -i ..//gcc/testsuite/gcc.target/i386/fentryname3.c
```

If you have installed additional packages such as valgrind and gdb, the gcc part of the test suite will run more tests than in LFS. Some of those will report FAIL and others XPASS (pass when expected to FAIL). As of gcc-14.1.0, about 74 FAILs occur in the “quality” suite, as well as miscellaneous failures throughout the rest of the test suite. If all the compilers above are built, there will be a little over 110 unexpected failures out of over 617,000 tests. To run the tests, issue:

```
make -k check
```

The tests are very long, and the results may be hard to find in the logs, specially if you use parallel jobs with make. You can get a summary of the tests with:

```
../contrib/test_summary
```

Now, as the `root` user:

```
make install &&

mkdir -pv /usr/share/gdb/auto-load/usr/lib      &&
mv -v /usr/lib/*gdb.py /usr/share/gdb/auto-load/usr/lib &&

chown -v -R root:root \
/usr/lib/gcc/*linux-gnu/14.2.0/include{,-fixed}     &&

ln -sfv ../../libexec/gcc/${(gcc -dumpmachine)}/14.2.0/liblto_plugin.so \
/usr/lib/bfd-plugins/
```

Command Explanations

`mkdir build; cd build`: The GCC documentation recommends building the package in a dedicated build directory.

`--disable-multilib`: This parameter ensures that files are created for the specific architecture of your computer.

`--with-system-zlib`: Uses the system zlib instead of the bundled one. zlib is used for compressing and decompressing GCC's intermediate language in LTO (Link Time Optimization) object files.

`--enable-default-pie`: Makes the `-fpie` option the default when compiling programs. Together with the [ASLR](#) feature enabled in the kernel, this defeats some kind of attacks based on known memory layouts.

`--enable-default-ssp`: Makes the `-fstack-protector-strong` option the default when compiling programs. [SSP](#) is a technique preventing alteration of the program flow by corrupting the parameter stack.

`--enable-host-pie`: Makes the compiler executables PIE (Position Independent Executable). This can be used to enhance protection against ROP (Return Oriented Programming) attacks, and can be viewed as part of a wider trend to harden binaries.

`--enable-languages=c,c++,fortran,go,objc,obj-c++,m2`: This command identifies which languages to build. You may modify this command to remove undesired languages. GCC also supports Ada and D, but building GCC with Ada (or D) support needs an existing Ada (or D) compiler. So they are not enabled here.

`make -k check`: This command runs the test suite without stopping if any errors are encountered.

`.../contrib/test_summary`: This command will produce a summary of the test suite results. You can append `| grep -A7 summ` to the command to produce an even more condensed version of the summary. You may also wish to redirect the output to a file for review and comparison later on.

`mv -v /usr/lib/*gdb.py ...`: The installation stage puts some files used by gdb under the `/usr/lib` directory. This generates spurious error messages when performing `ldconfig`. This command moves the files to another location.

`chown -v -R root:root /usr/lib/gcc/*linux-gnu/...`: If the package is built by a user other than root, the ownership of the installed `include` directory (and its content) will be incorrect. This command changes the ownership to the `root` user and group.

`--enable-host-shared --enable-languages=jit`: Build `libgccjit`, a library for embedding GCC inside programs and libraries for generating machine code. Despite "JIT" (just-in-time) in the name, the library can be used for AOT (ahead-of-time) compilation as well. `--enable-host-shared` is needed for building `libgccjit`, but it significantly slows down GCC. So `libgccjit` should be built and installed separately, not as a part of the "main" GCC installation. If you need this library, configure GCC with these two options and install the library by running `make -C gcc jit.install-common jit.install-info` as the `root` user. This library is not used by any BLFS package, nor tested by the BLFS developers.

Contents

Some program and library names and descriptions are not listed here, but can be found at [LFS section for GCC](#) as they were initially installed during the building of LFS.

Installed Programs: `gccgo`, `gfortran`, `gm2`, `go`, and `gofmt`, hard-linked to architecture specific names

Installed Libraries: `libgfortran.{so,a}`, `libgm2.{so,a}` `libgo.{so,a}`, `libgobegin.a`, `libgolibbegin.a`, `libobjc.{so,a}`, and numerous other run-time libraries and executables

Installed Directories: `/usr/lib/go`

Short Descriptions

<code>gccgo</code>	is a GCC-based compiler for the Go language
<code>gm2</code>	is a GCC-based compiler for the Modula-2 language
<code>go</code>	is a tool for managing Go source code
<code>gofmt</code>	is a tool for formatting Go source code
<code>gfortran</code>	is a GCC-based compiler for the Fortran language

GC-8.2.6

Introduction to GC

The GC package contains the Boehm-Demers-Weiser conservative garbage collector, which can be used as a garbage collecting replacement for the C `malloc` function or C++ `new` operator. It allows you to allocate memory basically as you normally would, without explicitly deallocating memory that is no longer useful. The collector automatically recycles memory when it determines that it can no longer be otherwise accessed. The collector is also used by a number of programming language implementations that either use C as intermediate code, want to facilitate easier interoperation with C libraries, or just prefer the simple collector interface. Alternatively, the garbage collector may be used as a leak detector for C or C++ programs, though that is not its primary goal.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/ivmai/bdwgc/releases/download/v8.2.6/gc-8.2.6.tar.gz>
- Download MD5 sum: fc5351214bc2e854070ee3319181a467
- Download size: 1.2 MB
- Estimated disk space required: 11 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Optional

[libatomic_ops-7.8.2](#)

Installation of GC

Install GC by running the following commands:

```
./configure --prefix=/usr      \
            --enable-cplusplus \
            --disable-static   \
            --docdir=/usr/share/doc/gc-8.2.6 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
install -v -m644 doc/gc.man /usr/share/man/man3/gc_malloc.3
```

Command Explanations

`--docdir=/usr/share/doc/gc-8.2.6`: This option is used so the package will install the documentation in a versioned directory.

`--enable-cplusplus`: This parameter enables the building and installing of the C++ library along with the standard C library.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libcord.so, libgc.so, libgccpp.so, and libgctba.so

Installed Directories: /usr/include/gc and /usr/share/doc/gc-8.2.6

Short Descriptions

libcord.so	contains a tree-based string library
libgc.so	contains a C interface to the conservative garbage collector, primarily designed to replace the C malloc function
libgccpp.so	contains a C++ interface to the conservative garbage collector
libgctba.so	contains a C++ interface to throw bad allocations

GDB-15.1

Introduction to GDB

GDB, the GNU Project debugger, allows you to see what is going on “inside” another program while it executes -- or what another program was doing at the moment it crashed. Note that GDB is most effective when tracing programs and libraries that were built with debugging symbols and not stripped.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/gdb/gdb-15.1.tar.xz>

- Download MD5 sum: 494e3beaac44e66367c3e443a4414529
- Download size: 23 MB
- Estimated disk space required: 806 MB (add 1.0 GB for docs; add 720 MB for tests)
- Estimated build time: 0.9 SBU (add 0.4 SBU for docs; see below for tests; all using parallelism=8)

GDB Dependencies

Recommended Runtime Dependency

[six-1.16.0](#) (Python 3 module, required at run-time to use GDB scripts from various LFS/BLFS packages with Python 3 installed in LFS)

Optional

[Doxygen-1.12.0](#), [GCC-14.2.0](#) (ada, gfortran, and go are used for tests), [Guile-3.0.10](#), [rustc-1.80.1](#) (used for some tests), [Valgrind-3.23.0](#), and [SystemTap](#) (run-time, used for tests)

Installation of GDB

Install GDB by running the following commands:

```
mkdir build &&
cd build &&

./configure --prefix=/usr      \
            --with-system-readline \
            --with-python=/usr/bin/python3 &&
make
```

Optionally, to build the API documentation using [Doxygen-1.12.0](#), run:

```
make -C gdb/doc doxy
```

Running the tests is not recommended. The results vary a lot depending on the system architecture and what optional dependencies are installed and what version of gcc is being used. On one system tested, there were 140 unexpected failures (out of over 108,000 tests) and on another system there were "only" 32 unexpected failures. The time to run the tests varies from approximately 6 SBU to over 15 SBU when using -j8. This depends on number of tests that time out as well as other factors.

Tip

With a plain `make check`, there are many warning messages about a missing global configuration file. These can be avoided by running `touch global.exp` and prepending the `make check` command with `DEJAGNU=$PWD/global.exp`. In addition the tests can be speeded up considerably by using the `make` option "`-j<N>`" where `<N>` is the number of cores on your system.

To test the results anyway, issue:

```
pushd gdb/testsuite &&
make site.exp      &&
echo "set gdb_test_timeout 30" >> site.exp &&
make check 2>1 | tee gdb-check.log
popd
```

See `gdb/testsuite/README` and [TestingGDB](#). There are many additional problems with the test suite:

- Clean directories are needed if re-running the tests. For that reason, make a copy of the compiled source code directory before the tests in case you need to run the tests again.
- Results depend on installed compilers.
- On some AMD-based systems, over 200 additional tests may fail due to a difference in the threading implementation on those CPUs.

Now, as the `root` user:

```
make -C gdb install &&
```

```
make -C gdbserver install
```

If you have built the API documentation, it is now in `gdb/doc/doxy`. You can install it (as the `root` user):

```
install -d /usr/share/doc/gdb-15.1 &&
rm -rf gdb/doc/doxy/xml &&
cp -Rv gdb/doc/doxy /usr/share/doc/gdb-15.1
```

Command Explanations

`--with-system-readline`: This switch forces GDB to use the copy of Readline installed in LFS.

`--with-python=/usr/bin/python3`: This switch forces GDB to use Python 3.

Contents

Installed Programs: `gcore`, `gdb`, `gdbserver`, and `gdb-add-index`

Installed Library: `libinproctrace.so`

Installed Directories: `/usr/{include,share}/gdb` and `/usr/share/doc/gdb-15.1`

Short Descriptions

<code>gcore</code>	generates a core dump of a running program
<code>gdb</code>	is the GNU Debugger
<code>gdbserver</code>	is a remote server for the GNU debugger (it allows programs to be debugged from a different machine)
<code>gdb-add-index</code>	Allows adding index files to ELF binaries. This speeds up <code>gdb</code> start on large programs.
<code>libinproctrace.so</code>	contains functions for the in-process tracing agent. The agent allows for installing fast tracepoints, listing static tracepoint markers, probing static tracepoints markers, and starting trace monitoring.

Git-2.46.0

Introduction to Git

Git is a free and open source, distributed version control system designed to handle everything from small to very large projects with speed and efficiency. Every Git clone is a full-fledged repository with complete history and full revision tracking capabilities, not dependent on network access or a central server. Branching and merging are fast and easy to do. Git is used for version control of files, much like tools such as [Mercurial-6.8.1](#), Bazaar, [Subversion-1.14.3](#), [CVS](#), Perforce, and Team Foundation Server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/software/scm/git/git-2.46.0.tar.xz>
- Download MD5 sum: 2309cd803a02378380f24c50667d9bfb
- Download size: 7.2 MB
- Estimated disk space required: 441 MB (with downloaded documentation; add 19 MB for building docs; add 21 MB for tests)
- Estimated build time: 0.3 SBU (with parallelism=4; add 0.4 SBU for building doc, and up to 7 SBU (disk speed dependent) for tests)

Additional Downloads

- <https://www.kernel.org/pub/software/scm/git/git-manpages-2.46.0.tar.xz> (not needed if you've installed [asciidoc-10.2.1](#), [xmlto-0.29](#), and prefer to rebuild them)
- <https://www.kernel.org/pub/software/scm/git/git-htmldocs-2.46.0.tar.xz> and other docs (not needed if you've installed [asciidoc-10.2.1](#) and want to rebuild the documentation).

Git Dependencies

Recommended

[cURL-8.9.1](#) (needed to use Git over http, https, ftp or ftps)

Optional

[Apache-2.4.62](#) (for some tests), [Fcron-3.2.1](#) (runtime, for scheduling `git maintenance` jobs), [GnuPG-2.4.5](#) (runtime, may be used to sign Git commits or tags, or verify the signatures of them), [OpenSSH-9.8p1](#) (runtime, needed to use Git over ssh), [pcre2-10.44](#), [Subversion-1.14.3](#) with Perl bindings (runtime, for `git svn`), [Tk-8.6.14](#) (gitk, a simple Git repository viewer, uses Tk at runtime), [Valgrind-3.23.0](#), [Authen::SASL](#) and [MIME::Base64](#) (both runtime, for `git send-email`), [IO-Socket-SSL-2.088](#) (runtime, for `git send-email` to connect to a SMTP server with SSL encryption), and [Systemd-256.4](#) (runtime, rebuilt with [Linux-PAM-1.6.1](#), for scheduling `git maintenance` jobs)

Optional (to create the man pages, html docs and other docs)

[xmlto-0.0.29](#) and [asciidoc-10.2.1](#), and also [dblatex](#) (for the PDF version of the user manual), and [docbook2x](#) to create info pages

Installation of Git

Install Git by running the following commands:

```
./configure --prefix=/usr \
            --with-gitconfig=/etc/gitconfig \
            --with-python=python3 &&
make
```

You can build the man pages and/or html docs, or use downloaded ones. If you choose to build them, use the next two instructions.

If you have installed [asciidoc-10.2.1](#) you can create the html version of the man pages and other docs:

```
make html
```

If you have installed [asciidoc-10.2.1](#) and [xmlto-0.0.29](#) you can create the man pages:

```
make man
```

The test suite can be run in parallel mode. To run the test suite, issue: `make test -k |& tee test.log`. If some test fails, the list of failed tests can be shown via `grep '^not ok' test.log | grep -v TODO`.

Now, as the `root` user:

```
make perllibdir=/usr/lib/perl5/5.40/site_perl install
```

If you created the man pages and/or html docs

Install the man pages as the `root` user:

```
make install-man
```

Install the html docs as the `root` user:

```
make htmlldir=/usr/share/doc/git-2.46.0 install-html
```

If you downloaded the man pages and/or html docs

If you downloaded the man pages untar them as the `root` user:

```
tar -xf ../git-manpages-2.46.0.tar.xz \
      -C /usr/share/man --no-same-owner --no-overwrite-dir
```

If you downloaded the html docs untar them as the `root` user:

```
mkdir -vp /usr/share/doc/git-2.46.0 &&
tar -xf ../git-htmldocs-2.46.0.tar.xz \
      -C /usr/share/doc/git-2.46.0 --no-same-owner --no-overwrite-dir &&

find      /usr/share/doc/git-2.46.0 -type d -exec chmod 755 {} \; &&
find      /usr/share/doc/git-2.46.0 -type f -exec chmod 644 {} \;
```

Reorganize text and html in the html-docs (both methods)

For both methods, the html-docs include a lot of plain text files. Reorganize the files as the `root` user:

```
mkdir -vp /usr/share/doc/git-2.46.0/man-pages/{html,text}      &&
mv      /usr/share/doc/git-2.46.0/{git*.txt,man-pages/text}      &&
mv      /usr/share/doc/git-2.46.0/{git*,index.,man-pages}/html &&

mkdir -vp /usr/share/doc/git-2.46.0/technical/{html,text}      &&
mv      /usr/share/doc/git-2.46.0/technical/{*.txt,text}      &&
mv      /usr/share/doc/git-2.46.0/technical/{*,}html          &&

mkdir -vp /usr/share/doc/git-2.46.0/howto/{html,text}      &&
mv      /usr/share/doc/git-2.46.0/howto/{*.txt,text}      &&
mv      /usr/share/doc/git-2.46.0/howto/{*,}html          &&

sed -i '/^<a href=/s|howto/|&html/|' /usr/share/doc/git-2.46.0/howto-index.html &&
sed -i '/^<link:/s|howto/|&html/|' /usr/share/doc/git-2.46.0/howto-index.txt
```

Command Explanations

`--with-gitconfig=/etc/gitconfig`: This sets `/etc/gitconfig` as the file that stores the default, system wide, Git settings.

`--with-python=python3`: Use this switch to use Python 3, instead of the EOL'ed Python 2. Python is used for the `git p4` interface to Perforce repositories, and also used in some tests.

`--with-libpcre2`: Use this switch if PCRE2 is installed.

`tar -xf ..git-manpages-2.46.0.tar.gz -C /usr/share/man --no-same-owner`: This untars `git-manpages-2.46.0.tar.gz`. The `-C` option makes tar change directory to `/usr/share/man` before it starts to decompress the docs. The `--no-same-owner` option stops tar from preserving the user and group details of the files. This is useful as that user or group may not exist on your system; this could (potentially) be a security risk.

`mv /usr/share/doc/git-2.46.0 ...`: These commands move some of the files into subfolders to make it easier to sort through the docs and find what you're looking for.

`find ... chmod ...`: These commands correct the permissions in the shipped documentation tar file.

Configuring Git

Config Files

`~/.gitconfig` and `/etc/gitconfig`

Contents

Installed Programs: git, git-receive-pack, git-upload-archive, and git-upload-pack (hardlinked to each other), git-cvsserver, git-shell, gitk, and scalar

Installed Libraries: None

Installed Directories: /usr/libexec/git-core, /usr/lib/perl5/5.40/site_perl/Git, and /usr/share/{doc/git-2.46.0,git-core,git-gui,gitk,gitweb}

Short Descriptions

<code>git</code>	is the stupid content tracker
<code>git-cvsserver</code>	is a CVS server emulator for Git
<code>gitk</code>	is a graphical Git repository browser (needs Tk-8.6.14)
<code>git-receive-pack</code>	is invoked by <code>git send-pack</code> and updates the repository with the information fed from the remote end
<code>git-shell</code>	is a login shell for SSH accounts to provide restricted Git access
<code>git-upload-archive</code>	is invoked by <code>git archive --remote</code> and sends a generated archive to the other end over the git protocol
<code>git-upload-pack</code>	is invoked by <code>git fetch-pack</code> , it discovers what objects the other side is missing, and sends them after packing
<code>scalar</code>	is a repository management tool that optimizes Git for use in large repositories

Running a Git Server

Introduction

This section will describe how to set up, administer and secure a git server. Git has many options available. For more detailed documentation see <https://git-scm.com/book/en/v2>.

Server Dependencies

Required

[git-2.46.0](#) and [OpenSSH-9.8p1](#)

Setting up a Git Server

The following instructions will install a git server. It will be set up to use OpenSSH as the secure remote access method.

Configuration of the server consists of the following steps:

1. Set Up Users, Groups, and Permissions

You will need to be user `root` for the initial portion of configuration. Create the `git` user and group and set and unusable password hash with the following commands:

```
groupadd -g 58 git &&
useradd -c "git Owner" -d /home/git -m -g git -s /usr/bin/git-shell -u 58 git &&
sed -i '/^git:/s/^git:[^:]*/git:NP:/' /etc/shadow
```

Putting in an unusable password hash (replacing the `!` by `NP`) unlocks the account but it cannot be used to login via password authentication. That is required by sshd to work properly. Next, create some files and directories in the home directory of the git user allowing access to the git repository using ssh keys.

```
install -o git -g git -dm0700 /home/git/.ssh &&
install -o git -g git -m0600 /dev/null /home/git/.ssh/authorized_keys
```

For any developer who should have access to the repository add his/her public ssh key to `/home/git/.ssh/authorized_keys`. First, prepend some options to prevent users from using the connection to git for port forwarding to other machines the git server might reach.

```
echo -n "no-port-forwarding,no-X11-forwarding,no-agent-forwarding,no-pty" >> /home/git/.ssh/authorized_keys &&
cat <user-ssh-key> >> /home/git/.ssh/authorized_keys
```

It is also useful to set the default name of the initial branch of new repositories by modifying the git configuration. As the `root` user, run:

```
git config --system init.defaultBranch trunk
```

Finally add the `/usr/bin/git-shell` entry to the `/etc/shells` configuration file. This shell has been set in the `git` user profile and is to make sure that only git related actions can be executed:

```
echo "/usr/bin/git-shell" >> /etc/shells
```

2. Create a git repository

The repository can be anywhere on the filesystem. It is important that the git user has read/write access to that location. We use `/srv/git` as base directory. Create a new git repository with the following commands (as the `root` user):

Note

In all the instructions below, we use `project1` as an example repository name. You should name your repository as a short descriptive name for your specific project.

```
install -o git -g git -m755 -d /srv/git/project1.git &&
cd /srv/git/project1.git &&
git init --bare &&
chown -R git:git .
```

3. Populate the repository from a client system

Note

All the instructions in this section and the next should be done on a user system, not the server system.

Now that the repository is created, it can be used by the developers to put some files into it. Once the ssh key of the user is imported to git's `authorized_keys` file, the user can interact with the repository.

A minimal configuration should be available on the developer's system specifying its user name and the email address. Create this minimal config file on client side:

```
cat > ~/.gitconfig <<EOF
[user]
    name = <users-name>
    email = <users-email-address>
EOF
```

On the developer's machine, set up some files to be pushed to the repository as the initial content:

Note

The `gitserver` term used below should be the host name (or ip address) of the git server.

```
mkdir myproject
cd myproject
git init --initial-branch=trunk
git remote add origin git@gitserver:/srv/git/project1.git
cat >README <<EOF
This is the README file
EOF
git add README
git commit -m 'Initial creation of README'
git push --set-upstream origin trunk
```

The initial content is now pushed to the server and is available for other users. On the current machine, the argument `--set-upstream origin trunk` is now no longer required as the local repository is now connected to the remote repository. Subsequent pushes can be performed as

```
git push
```

Other developers can now clone the repository and do modifications to the content (as long as their ssh keys has been installed):

```
git clone git@gitserver:/srv/git/project1.git
cd project1
vi README
git commit -am 'Fix for README file'
git push
```

Note

This is a very basic server setup based on OpenSSH access. All developers are using the `git` user to perform actions on the repository and the changes users are committing can be distinguished as the local user name (see `~/.gitconfig`) is recorded in the changesets.

Access is restricted by the public keys added to git's `authorized_keys` file and there is no option for the public to export/clone the repository. To enable this, continue with step 4 to set up the git server for public read-only access.

In the URL used to clone the project, the absolute path (here `/srv/git/project1.git`) has to be specified as the repository is not in git's home directory but in `/srv/git`. To get rid of the need to expose the structure of the server installation, a symlink can be added in git's home directory for each project like this:

```
ln -svf /srv/git/project1.git /home/git/
```

Now, the repository can be cloned using

```
git clone git@gitserver:project1.git
```

4. Configure the Server

The setup described above makes a repository available for authenticated users (via providing the ssh public key file). There is also a simple way to publish the repository to unauthenticated users — of course without write access.

The combination of access via ssh (for authenticated users) and the export of repositories to unauthenticated users via the daemon is in most cases enough for a development site.

Note

The daemon will be reachable at port 9418 by default. Make sure that your firewall setup allows access to that port.

To start the server at boot time, install the `git-daemon.service` unit from the [blfs-systemd-units-20240801](#) package:

```
make install-git-daemon
```

In order to allow git to export a repository, a file named `git-daemon-export-ok` is required in each repository directory on the server. The file needs no content, just its existence enables its absence disables the export of that repository.

```
touch /srv/git/project1.git/git-daemon-export-ok
```

Along with the `git-daemon.service` unit, a configuration file named `/etc/default/git-daemon` has been installed. Review this configuration file to match your needs.

There are only three options to set in the configuration file:

- `GIT_BASE_DIR=<dirname>`

Specify the location of the git repositories. Relative paths used when accessing the daemon will be translated relative to this directory.

- `DFT_REPO_DIR=<dirname>`

This directory is added to the white list of allowed directories. This variable can hold multiple directory names but is usually set equal to `GIT_BASE_DIR`.

- `GIT_DAEMON_OPTS=<options>`

In case special options to the `git daemon` command are needed, they have to be specified in this setting. One example might be to adjust the port number where the daemon is listening. In this case, add `--port=<port number>` to this variable. For more information about which options can be set, take a look at the output of `git daemon --help`.

After starting the daemon, unauthenticated users can clone exported repositories by using

```
git clone git://gitserver/project1.git
```

As the base directory is `/srv/git` by default (or set to a custom value in the configuration), git interprets the incoming path (`/project1.git`) relative to that base directory so that the repository in `/srv/git/project1.git` is served.

Guile-3.0.10

Introduction to Guile

The Guile package contains the GNU Project's extension language library. Guile also contains a stand alone Scheme interpreter.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/guile/guile-3.0.10.tar.xz>
- Download MD5 sum: 375f8a423a23d229552512113aa4a954

- Download size: 5.5 MB
- Estimated disk space required: 184 MB (add 4 MB for tests)
- Estimated build time: 4.5 SBU (Using parallelism=4; add 0.3 SBU for tests)

Guile Dependencies

Required

[GC-8.2.6](#) and [libunistring-1.2](#)

Optional

[Emacs-29.4](#) and [GDB-15.1](#) (run-time only dependencies).

Installation of Guile

Install Guile by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --docdir=/usr/share/doc/guile-3.0.10 &&
make      &&
make html &&

makeinfo --plaintext -o doc/r5rs/r5rs.txt doc/r5rs/r5rs.texi &&
makeinfo --plaintext -o doc/ref/guile.txt doc/ref/guile.texi
```

To test the results, issue: `./check-guile`. On an i686 system, as many as 10 tests will fail in the `numbers.test` test suite due to minor floating-point rounding errors.

Now, as the `root` user:

```
make install      &&
make install-html &&

mkdir -p          /usr/share/gdb/auto-load/usr/lib &&
mv /usr/lib/libguile-*-gdb.scm /usr/share/gdb/auto-load/usr/lib &&
mv /usr/share/doc/guile-3.0.10/{guile.html,ref} &&
mv /usr/share/doc/guile-3.0.10/r5rs{.html,}      &&

find examples -name "Makefile*" -delete      &&
cp -vR examples   /usr/share/doc/guile-3.0.10 &&

for dirname in r5rs ref; do
    install -v -m644 doc/${dirname}/*.txt \
                /usr/share/doc/guile-3.0.10/${dirname}
done &&
unset dirname
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: guile, guile, guile-config, guile-snarf and guile-tools

Installed Libraries: libguile-3.0.so and guile-readline.so

Installed Directories: /usr/include/guile, /usr/lib/guile, /usr/share/doc/guile-3.0.10 and /usr/share/guile

Short Descriptions

<code>guile</code>	is a stand-alone Scheme interpreter for Guile
<code>guile-</code> <code>config</code>	is a Guile script which provides the information necessary to link your programs against the Guile library, in much the same way PkgConfig does
<code>guile-</code> <code>snarf</code>	is a script to parse declarations in your C code for Scheme visible C functions

`guild` is a wrapper program installed along with `guile`, which knows where a particular module is installed and calls it, passing its arguments to the program

`guile-tools` is a symlink to `guild`

LLVM-18.1.7

Introduction to LLVM

The LLVM package contains a collection of modular and reusable compiler and toolchain technologies. The Low Level Virtual Machine (LLVM) Core libraries provide a modern source and target-independent optimizer, along with code generation support for many popular CPUs (as well as some less common ones!). These libraries are built around a well specified code representation known as the LLVM intermediate representation ("LLVM IR").

Clang provides new C, C++, Objective C and Objective C++ front-ends for LLVM and is required by some desktop packages such as firefox and for rust if that is built using the system LLVM.

The Compiler RT package provides runtime sanitizer and profiling libraries for developers who use Clang and LLVM.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/llvm/llvm-project/releases/download/llvmorg-18.1.7/llvm-18.1.7.src.tar.xz>
- Download MD5 sum: ad39785449a878df1eed590339c3a8c4
- Download size: 60 MB
- Estimated disk space required: 3.5 GB (964 MB installed; add 20 GB for tests and 308 MB for documentation)
- Estimated build time: 13 SBU (Add 8 SBU for tests; both using parallelism=8)

Additional Downloads

Cmake modules for LLVM

- Download: <https://anduin.linuxfromscratch.org/BLFS/llvm/llvm-cmake-18.src.tar.xz>
- Download MD5 sum: bf2ad617d47ce40fe77c0e5c26b1fe43
- Download size: 12 KB

Third-party dependencies for LLVM build system

- Download: <https://anduin.linuxfromscratch.org/BLFS/llvm/llvm-third-party-18.src.tar.xz>
- Download MD5 sum: 5ebac19868c66cdac8b87077faefd38c
- Download size: 396 KB

Recommended Download

Clang

- Download: <https://github.com/llvm/llvm-project/releases/download/llvmorg-18.1.7/clang-18.1.7.src.tar.xz>
- Download MD5 sum: 47e26237223d5ee0e6b674f028b56dae
- Download size: 22 MB

Optional Download

Compiler RT

- Download: <https://github.com/llvm/llvm-project/releases/download/llvmorg-18.1.7/compiler-rt-18.1.7.src.tar.xz>
- Download MD5 sum: 31753c0ec6271a37c12a17cf84187098
- Download size: 2.4 MB

LLVM Dependencies

Required

[CMake-3.30.2](#)

Optional

[Doxygen-1.12.0](#), [git-2.46.0](#), [Graphviz-12.1.0](#), [libxml2-2.13.3](#), [psutil-6.0.0](#) (for tests), [Pygments-2.18.0](#), [PyYAML-6.0.2](#) (for tests), [rsync-3.3.0](#) (for tests), [Systemd-256.4](#) (rebuilt with PAM, for tests), [texlive-20240312](#) (or [install-tl-unx](#)), [Valgrind-3.23.0](#), [Zip-3.0](#), [myst-parser](#) (for building documentation), [OCaml](#), and [Z3](#)

Installation of LLVM

Two additional tarballs `llvm-cmake-18.src.tar.xz` and `llvm-third-party-18.src.tar.xz` are needed by LLVM building system. The upstream expects them extracted at the same level as the `llvm-18.1.7.src.tar.xz` tarball, and the extracted directories renamed to `cmake` and `third-party`. Extract them and modify the build system to avoid creating ambiguously-named directories outside the `llvm-18.1.7.src` hierarchy:

```
tar -xf .. llvm-cmake-18.src.tar.xz &&
tar -xf .. llvm-third-party-18.src.tar.xz &&
sed '/LLVM_COMMON_CMAKE_UTILS/s@.. cmake@llvm-cmake-18.src@' \
    -i CMakeLists.txt &&
sed '/LLVM_THIRD_PARTY_DIR/s@.. third-party@llvm-third-party-18.src@' \
    -i cmake/modules/HandleLLVMOPTIONS.cmake
```

Install clang into the source tree by running the following commands:

```
tar -xf .. clang-18.1.7.src.tar.xz -C tools &&
mv tools/clang-18.1.7.src tools/clang
```

If you have downloaded compiler-rt, install it into the source tree by running the following commands:

```
tar -xf .. compiler-rt-18.1.7.src.tar.xz -C projects &&
mv projects/compiler-rt-18.1.7.src projects/compiler-rt
```

There are many Python scripts in this package which use `/usr/bin/env python` to access the system Python which on LFS is [Python-3.12.5](#). Use the following command to fix these scripts:

```
grep -rl '#!*python' | xargs sed -i 's/python$/python3/'
```

Ensure installing the `FileCheck` program which is needed by the test suite of some packages (for example [rustc-1.80.1](#)):

```
sed 's/utility/tool/' -i utils/FileCheck/CMakeLists.txt
```

Install LLVM by running the following commands:

```
mkdir -v build &&
cd build &&

CC=gcc CXX=g++ \
cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_SKIP_INSTALL_RPATH=ON \
      -D LLVM_ENABLE_FFI=ON \
      -D CMAKE_BUILD_TYPE=Release \
      -D LLVM_BUILD_LLVM_DYLIB=ON \
      -D LLVM_LINK_LLVM_DYLIB=ON \
      -D LLVM_ENABLE_RTTI=ON \
      -D LLVM_TARGETS_TO_BUILD="host;AMDGPU" \
      -D LLVM_BINUTILS_INCDIR=/usr/include \
      -D LLVM_INCLUDE_BENCHMARKS=OFF \
      -D CLANG_DEFAULT_PIE_ON_LINUX=ON \
      -D CLANG_CONFIG_FILE_SYSTEM_DIR=/etc/clang \
      -W no-dev -G Ninja ... &&
ninja
```

If you wish to run the tests, remove a test case known to hang indefinitely on some systems:

```
rm -f .. projects/compiler-rt/test/tsan/getline_nohang.cpp
```

LLVM test suite can produce many core dump files. They will occupy a large amount of disk space, and the core dump process can significantly slow down the testing. To test the results with core dump disabled, ensure [Systemd-256.4](#) and [Shadow-4.16.0](#) have been rebuilt with [Linux-PAM-1.6.1](#) support (if you are interacting via a SSH or graphical session, also ensure the [OpenSSH-9.8p1](#) server or the desktop manager has been built with [Linux-PAM-1.6.1](#)) and the current login session is started after updating the `/etc/pam.d/system-session` file to include `pam_systemd.so`, then issue:

```
systemctl --user start dbus &&
systemd-run --user --pty -d -G -p LimitCORE=0 ninja check-all
```

If `-jN` (N replaced with a number) is passed to `ninja`, the tests will be built with N logical cores, but run using all available logical cores. Run the test command in a cgroup (pass the `-p AllowedCPUs=...` option to the `systemd-run` command, read [the section called "Use Linux Control Group to Limit the Resource Usage"](#) for details) to limit the number of logical cores for running the tests. Two tests related to `TestCases/Linux/printf_fortify-5.c` are known to fail with Glibc-2.40 or newer. One test named `Linux/clone_setns.cpp` will fail if `CONFIG_USER_NS` is not enabled in kernel configuration.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D LLVM_ENABLE_FFI=ON`: This switch allows LLVM to use libffi.

`-D LLVM_BUILD_LLVM_DYLIB=ON`: This switch builds the libraries as static and links all of them into an unique shared one. This is the recommended way of building a shared library.

`-D CMAKE_BUILD_TYPE=Release`: This switch enables compiler optimizations in order to speed up the code and reduce its size. It also disables some compile checks which are not necessary on a production system.

`-D LLVM_TARGETS_TO_BUILD="host;AMDGPU"`: This switch enables building for the same target as the host, and also for the r600 AMD GPU used by the Mesa r600 and radeonsi drivers. The default is all of the targets. You can use a semicolon separated list. Valid targets are: host, AArch64, AMDGPU, ARM, AVR, BPF, Hexagon, Lanai, LoongArch, Mips, MSP430, NVPTX, PowerPC, RISCV, Sparc, SystemZ, SystemZ, VE, WebAssembly, X86, XCore, or all.

`-D LLVM_LINK_LLVM_DYLIB=ON`: Used in conjunction with `-D LLVM_BUILD_LLVM_DYLIB=ON`, this switch enables linking the tools against the shared library instead of the static ones. It slightly reduces their size and also ensures that `llvm-config` will correctly use `libLLVM-18.so`.

`-D LLVM_ENABLE_RTTI=ON`: This switch is used to build LLVM with run-time type information. This is required for building [Mesa-24.1.5](#).

`-D LLVM_BINUTILS_INCDIR=/usr/include`: This switch is used to tell the build system the location of binutils headers, which were installed in LFS. This allows the building of `LLVMgold.so`, which is needed for building programs with `clang` and Link Time Optimization (LTO).

`-D LLVM_INCLUDE_BENCHMARKS=OFF`: is used to disable generation build targets for the LLVM benchmarks. This option requires additional code that is not currently available.

`-D CLANG_DEFAULT_PIE_ON_LINUX=ON`: makes `-fPIE` option the default when compiling programs. Together with the [ASLR](#) feature enabled in the kernel, this defeats some kind of attacks based on known memory layouts.

`-D CLANG_CONFIG_FILE_SYSTEM_DIR=/etc/clang`: makes `clang` and `clang++` search `/etc/clang` for configuration files.

`-D BUILD_SHARED_LIBS=ON`: if used instead of `-D LLVM_BUILD_LLVM_DYLIB=ON` and `-D LLVM_LINK_LLVM_DYLIB=ON`, builds all the LLVM libraries (about 60) as shared libraries instead of static.

`-D LLVM_ENABLE_DOXYGEN`: Enables the generation of browsable HTML documentation if you have installed [Doxygen-1.12.0](#). You should run `make doxygen-html` afterwards, and install the generated documentation manually.

Configuring LLVM

Configuration Information

If you've built Clang, as the `root` user create two configuration files to make [SSP](#) enabled by default for `clang` and `clang++`, so the default configuration of their SSP feature will be consistent with [GCC-14.2.0](#):

```
mkdir -pv /etc/clang &&
for i in clang clang++; do
    echo -fstack-protector-strong > /etc/clang/$i.cfg
done
```

Contents

Installed Programs: amdgpu-arch, analyze-build, bugpoint, c-index-test, clang, clang++ (symlinks to clang-18), clang-18, clang-check, clang-cl, clang-cpp (last two symlinks to clang), clang-extdef-mapping, clang-format, clang-linker-wrapper, clang-offload-bundler, clang-offload-packager, clang-refactor, clang-rename, clang-repl, clang-scan-deps, clang-tblgen, diagtool, dsymutil, FileCheck, git-clang-format, hmaptool, intercept-build, llc, lli, llvm-addr2line (symlink to llvm-symbolizer), llvm-ar, llvm-as, llvm-bcanalyzer, llvm-bitcode-strip (symlink to llvm-objcopy), llvm-cat, llvm-cfi-verify, llvm-config, llvm-cov, llvm-c-test, llvm-cvtres, llvm-cxxdump, llvm-cxxfilt, llvm-cxxmap, llvm-debuginfo-analyzer, llvm-debuginfod, llvm-debuginfod-find, llvm-diff, llvm-dis, llvm-dltool (symlink to llvm-ar), llvm-dwarfddump, llvm-dwarfutil, llvm-dwp, llvm-exegesis, llvm-extract, llvm-gsymutil, llvm-ifs, llvm-install-name-tool (symlink to llvm-objcopy), llvm-jitlink, llvm-lib (symlink to llvm-ar), llvm-libtool-darwin, llvm-link, llvm-lipo, llvm-lto, llvm-lto2, llvm-mc, llvm-mca, llvm-ml, llvm-modextract, llvm-mt, llvm-nm, llvm-objcopy, llvm-objdump, llvm-opt-report, llvm-otool (symlink to llvm-objdump), llvm-poutil, llvm-propdata, llvm-profgen, llvm-ranlib (symlink to llvm-ar), llvm-rc, llvm-readelf (symlink to llvm-readobj), llvm-readobj, llvm-readtapi, llvm-reduce, llvm-remarkutil, llvm-rtdyld, llvm-sim, llvm-size, llvm-split, llvm-stress, llvm-strings, llvm-strip (symlink to llvm-objcopy), llvm-symbolizer, llvm-tblgen, llvm-tli-checker, llvm-undname, llvm-windres (symlink to llvm-rc), llvm-xray, nvptx-arch, opt, sancov, sanstats, scan-build, scan-build-py, scan-view, and verify-uselistorder

Installed Libraries: libLLVM.so, libLLVM*.a (100 libraries), libLTO.so, libRemarks.so, libclang.so, libclang-cpp.so, libclang*.a (42 libraries), and LLVMgold.so

Installed Directories: /usr/include/{clang,clang-c,llvm,llvm-c}, /usr/lib/{clang,cmake/{clang,llvm},libbear,libscanbuild}, /usr/share/{clang,opt-viewer,scan-build,scan-view}, and /etc/clang

Short Descriptions

amdgpu-arch	lists AMD GPUs installed; at runtime it needs <code>libhsa-runtime64.so</code> which is not a part of BLFS
analyze-build	is a static analysis tool
bugpoint	is the automatic test case reduction tool
c-index-test	is used to test the libclang API and demonstrate its usage
clang	is the Clang C, C++, and Objective-C compiler
clang-check	is a tool to perform static code analysis and display Abstract Syntax Trees (AST)
clang-extdef-mapping	is a tool to collect the USR name and location of external definitions in a source file
clang-format	is a tool to format C/C+/Java/JavaScript/Objective-C/Protobuf code
clang-linker-wrapper	is a wrapper utility over the host linker
clang-offload-bundler	is a tool to bundle/unbundle OpenMP offloaded files associated with a common source file
clang-offload-packager	is a tool to bundle several object files into a single binary, which can then be used to create a fatbinary containing offloading code
clang-refactor	is a Clang-based refactoring tool for C, C++ and Objective-C
clang-rename	is a tool to rename symbols in C/C++ programs
clang-scan-deps	is a tool to scan for dependencies in a source file
clang-tblgen	is a program that translates compiler-related target description (<code>.td</code>) files into C++ code and other output formats
diagtool	is a combination of tools for dealing with diagnostics in clang
FileCheck	is a tool that reads two files (one from standard input, and one specified on the command line) and uses one to verify the other.
dsymutil	is a tool used to manipulate archived DWARF debug symbol files, compatible with the Darwin command <code>dsymutil</code>
git-clang-format	runs clang-format on git generated patches (requires git-2.46.0)
hmaptool	is a Python tool to dump and construct header maps
intercept-build	generates a database of build commands for a project
llc	is the LLVM static compiler
lli	is used to directly execute programs from LLVM bitcode
llvm-addr2line	is a tool used to convert addresses into file names and line numbers
llvm-ar	is the LLVM archiver
llvm-as	is the LLVM assembler
llvm-bcanalyzer	is the LLVM bitcode analyzer
llvm-bitcode-strip	strips LLVM bitcode from an object
llvm-cat	is a tool to concatenate llvm modules

<code>llvm-cfi-verify</code>	identifies whether Control Flow Integrity protects all indirect control flow instructions in the provided object file, DSO, or binary
<code>llvm-config</code>	Prints LLVM compilation options
<code>llvm-cov</code>	is used to emit coverage information
<code>llvm-c-test</code>	is a bytecode disassembler
<code>llvm-cvtres</code>	is a tool to convert Microsoft resource files to COFF
<code>llvm-cxxdump</code>	is used as a C++ ABI Data Dumper
<code>llvm-cxxfilt</code>	is used to demangle C++ symbols in LLVM code
<code>llvm-cxxmap</code>	is used to remap C++ mangled symbols
<code>llvm-debuginfo-analyzer</code>	prints a logical representation of low-level debug information
<code>llvm-debuginfod</code>	is a service providing debug information over an HTTP API for analyzing stripped binaries
<code>llvm-debuginfod-find</code>	is an interface to the <code>llvm-debuginfod</code> daemon for finding debuginfod artifacts
<code>llvm-diff</code>	is the LLVM structural ' <code>diff</code> '
<code>llvm-dis</code>	is the LLVM disassembler
<code>llvm-dwarfdump</code>	prints the content of DWARF sections in object files
<code>llvm-dwarfutil</code>	is a tool to copy and manipulate debug info
<code>llvm-dwp</code>	merges split DWARF files
<code>llvm-elfabi</code>	is used to read information about an ELF binary's ABI
<code>llvm-exegesis</code>	is a benchmarking tool that uses information available in LLVM to measure host machine instruction characteristics like latency or port decomposition
<code>llvm-extract</code>	is used to extract a function from an LLVM module
<code>llvm-gsymutil</code>	is used to process GSYM Symbolication Format files which convert memory addresses to function name and source file line. These files are smaller than DWARF or Breakpad files
<code>llvm-ifcs</code>	is used to merge interface stubs with object files
<code>llvm-install-name-tool</code>	is used to rewrite load commands into MachO binary format
<code>llvm-jitlink</code>	is used to parse relocatable object files to make their contents executable in a target process
<code>llvm-libtool-darwin</code>	provides basic libtool functionality on Darwin-based systems. This is mostly useful if you are generating binaries for macOS systems
<code>llvm-link</code>	is the LLVM linker
<code>llvm-lipo</code>	is used to create universal binaries from MachO files
<code>llvm-lto</code>	is the LLVM LTO (link time optimization) linker
<code>llvm-lto2</code>	is a test harness for the resolution based LTO interface
<code>llvm-mc</code>	is a standalone machine code assembler/disassembler
<code>llvm-mca</code>	is a performance analysis tool to statically measure the performance of machine code
<code>llvm-ml</code>	is a playground for machine code provided by LLVM
<code>llvm-modextract</code>	is a tool to extract one module from multimodule bitcode files
<code>llvm-mt</code>	is a tool to generate signed files and catalogs from a side-by-side assembly manifest (used for Microsoft SDK)
<code>llvm-nm</code>	is used to list LLVM bitcode and object file's symbol table
<code>llvm-objcopy</code>	is LLVM's version of an objcopy tool
<code>llvm-objdump</code>	is an LLVM object file dumper
<code>llvm-opt-report</code>	is a tool to generate an optimization report from YAML optimization record files
<code>llvm-pdbutil</code>	is a PDB (Program Database) dumper. PDB is a Microsoft format
<code>llvm-profdata</code>	is a small tool to manipulate and print profile data files
<code>llvm-profgen</code>	generates LLVM SPGO profiling information
<code>llvm-ranlib</code>	is used to generate an index for a LLVM archive
<code>llvm-rc</code>	is a platform-independent tool to compile resource scripts into binary resource files
<code>llvm-readobj</code>	displays low-level format-specific information about object files
<code>llvm-readtapi</code>	is the LLVM TAPI file reader and transformer
<code>llvm-reduce</code>	is used to automatically reduce testcases when running a test suite
<code>llvm-remarkutil</code>	converts remark files between bitstream and YAML; or prints function instruction count information in remark files
<code>llvm-rtdyld</code>	is the LLVM MC-JIT tool

<code>llvm-size</code>	is the LLVM object size dumper
<code>llvm-split</code>	is the LLVM module splitter
<code>llvm-stress</code>	is used to generate random .ll files
<code>llvm-strings</code>	print strings found in a binary (object file, executable, or archive library)
<code>llvm-symbolizer</code>	converts addresses into source code locations
<code>llvm-tblgen</code>	is the LLVM Target Description To C++ Code Generator
<code>llvm-tli-checker</code>	is the LLVM TargetLibraryInfo versus SDK checker
<code>llvm-undname</code>	is a tool to demangle names
<code>llvm-xray</code>	is an implementation of Google's XRay function call tracing system
<code>nvptx-arch</code>	lists NVIDIA GPUs installed; at runtime it needs <code>libcuda.so</code> which is not a part of BLFS
<code>opt</code>	is the LLVM optimizer
<code>sancov</code>	is the sanitizer coverage processing tool
<code>sanstats</code>	is the sanitizer statistics processing tool
<code>scan-build</code>	is a Perl script that invokes the Clang static analyzer
<code>scan-build-py</code>	is a Python script that invokes the Clang static analyzer
<code>scan-view</code>	is a viewer for Clang static analyzer results
<code>verify-uselistorder</code>	is the LLVM tool to verify use-list order

Lua-5.4.7

Introduction to Lua

Lua is a powerful light-weight programming language designed for extending applications. It is also frequently used as a general-purpose, stand-alone language. Lua is implemented as a small library of C functions, written in ANSI C, and compiles unmodified in all known platforms. The implementation goals are simplicity, efficiency, portability, and low embedding cost. The result is a fast language engine with small footprint, making it ideal in embedded systems too.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.lua.org/ftp/lua-5.4.7.tar.gz>
- Download MD5 sum: fc3f3291353bbe6ee6dec85ee61331e8
- Download size: 368 KB
- Estimated disk space required: 3.8 MB (with Basic tests)
- Estimated build time: less than 0.1 SBU (with Basic tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/lua-5.4.7-shared_library-1.patch
- Optional Test Suite Download (HTTP): <https://www.lua.org/tests/lua-5.4.7-tests.tar.gz>
- Optional Test Suite Download MD5 sum: 5fcf34336004f83de447c79958ea678e
- Optional Test Suite Download size: 134 KB

Installation of Lua

Some packages check for the pkg-config file for Lua, which is created with:

```
cat > lua.pc << "EOF"
V=5.4
R=5.4.7

prefix=/usr
INSTALL_BIN=${prefix}/bin
INSTALL_INC=${prefix}/include
INSTALL_LIB=${prefix}/lib
INSTALL_MAN=${prefix}/share/man/man1
INSTALL_LMOD=${prefix}/share/lua/${V}
INSTALL_CMOD=${prefix}/lib/lua/${V}
exec_prefix=${prefix}
```

```

libdir=${exec_prefix}/lib
includedir=${prefix}/include

Name: Lua
Description: An Extensible Extension Language
Version: ${R}
Requires:
Libs: -L${libdir} -llua -lm -ldl
Cflags: -I${includedir}
EOF

```

Install Lua by running the following commands:

```

patch -Np1 -i ../lua-5.4.7-shared_library-1.patch &&
make linux

```

To test the results, issue: `make test`. "This will run the interpreter and print its version." More comprehensive tests can be performed if you downloaded the "Test suite" tarball. Those tests need to be executed after the package is installed, thus we defer to the description below.

Now, as the `root` user:

```

make INSTALL_TOP=/usr \
      INSTALL_DATA="cp -d" \
      INSTALL_MAN=/usr/share/man/man1 \
      TO_LIB="liblua.so liblua.so.5.4 liblua.so.5.4.7" \
      install &&

mkdir -pv /usr/share/doc/lua-5.4.7 &&
cp -v doc/*.{html,css,gif,png} /usr/share/doc/lua-5.4.7 &&

install -v -m644 -D lua.pc /usr/lib/pkgconfig/lua.pc

```

Here we describe only the "Basic tests". Untar the tarball and change to the `lua-5.4.7-tests` directory, then issue `lua -e "_U=true" all.lua`. If the tests finish without error, you will see a message containing the string "final OK".

Contents

Installed Programs: lua and luac

Installed Library: liblua.so

Installed Directories: /usr/{lib,share}/lua and /usr/share/doc/lua-5.4.7

Short Descriptions

<code>lua</code>	is the standalone Lua interpreter
<code>luac</code>	is the Lua compiler
<code>liblua.so</code>	contains the Lua API functions

Lua-5.2.4

Introduction to Lua 5.2

Lua is a powerful light-weight programming language designed for extending applications. It is also frequently used as a general-purpose, stand-alone language. Lua is implemented as a small library of C functions, written in ANSI C, and compiles unmodified in all known platforms. The implementation goals are simplicity, efficiency, portability, and low embedding cost. The result is a fast language engine with small footprint, making it ideal in embedded systems too.

This is an older version of Lua needed only for compatibility with other programs such as [Wireshark-4.2.6](#) and [VLC-3.0.21](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.lua.org/ftp/lua-5.2.4.tar.gz>
- Download MD5 sum: 913fdb32207046b273fdb17aad70be13
- Download size: 248 KB
- Estimated disk space required: 3.6 MB

- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/lua-5.2.4-shared_library-1.patch

Installation of Lua 5.2

Some packages check for the pkg-config file for Lua, which is created with:

```
cat > lua.pc << "EOF"
V=5.2
R=5.2.4

prefix=/usr
INSTALL_BIN=${prefix}/bin
INSTALL_INC=${prefix}/include/lua5.2
INSTALL_LIB=${prefix}/lib
INSTALL_MAN=${prefix}/share/man/man1
INSTALL_LMOD=${prefix}/share/lua/${V}
INSTALL_CMOD=${prefix}/lib/lua/${V}
exec_prefix=${prefix}
libdir=${exec_prefix}/lib
includedir=${prefix}/include/lua5.2

Name: Lua
Description: An Extensible Extension Language
Version: ${R}
Requires:
Libs: -L${libdir} -llu5.2 -lm -ldl
Cflags: -I${includedir}
EOF
```

Install Lua by running the following commands:

```
patch -Np1 -i ../lua-5.2.4-shared_library-1.patch &&
sed -i '/#define LUA_ROOT/s:/usr/local/:/usr:.' src/luacnf.h &&
sed -r -e '/^LU_(SO|A|T)=/ s/lua/lua5.2/' \
-e '/^LUAC_T=/ s/luac/luac5.2/' \
-i src/Makefile &&
make MYCFLAGS="-fPIC" linux
```

The installation of this package is complex, so we will use the DESTDIR method of installation:

```
make TO_BIN='lua5.2 luac5.2' \
TO_LIB="liblua5.2.so liblua5.2.so.5.2 liblua5.2.so.5.2.4" \
INSTALL_DATA="cp -d" \
INSTALL_TOP=$PWD/install/usr \
INSTALL_INC=$PWD/install/usr/include/lua5.2 \
INSTALL_MAN=$PWD/install/usr/share/man/man1 \
install &&
install -Dm644 lua.pc install/usr/lib/pkgconfig/lua52.pc &&
mkdir -pv install/usr/share/doc/lua-5.2.4 &&
cp -v doc/*.{html,css,gif,png} install/usr/share/doc/lua-5.2.4 &&
ln -s liblua5.2.so install/usr/lib/liblua.so.5.2 &&
ln -s liblua5.2.so install/usr/lib/liblua.so.5.2.4 &&
mv install/usr/share/man/man1/{lua.1,luac5.2.1} &&
mv install/usr/share/man/man1/{luac.1,luac5.2.1}
```

Now, as the `root` user:

```
chown -R root:root install &&
cp -a install/* /
```

Command Explanations

```
sed -i ... src/luac.h: This command changes the Lua search path to match the install path.
```

```
sed -i ... src/Makefile: This command deconflicts this installation with the latest version of lua.
```

Contents

Installed Programs: lua5.2 and luac5.2

Installed Library: liblua5.2.so

Installed Directories: /usr/include/lua5.2, /usr/lib/lua/5.2, /usr/share/doc/lua-5.2.4, and /usr/share/lua/5.2

Short Descriptions

- lua5.2 is the standalone Lua version 5.2 interpreter
- luac5.2 is the Lua version 5.2 compiler
- liblua5.2.so contains the Lua version 5.2 API functions

Mercurial-6.8.1

Introduction to Mercurial

Mercurial is a distributed source control management tool similar to Git and Bazaar. Mercurial is written in Python and is used by projects such as Mozilla for Firefox and Thunderbird.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.mercurial-scm.org/release/mercurial-6.8.1.tar.gz>
- Download MD5 sum: f0c076d22c07bf65ffb02753fc516084
- Download size: 7.9 MB
- Estimated disk space required: 113 MB (with docs, add 1.5 GB for tests)
- Estimated build time: 0.3 SBU (with docs; add 16 SBU for tests; both using parallelism=8)

Mercurial Dependencies

Optional

[docutils-0.21.2](#) (required to build the documentation), [git-2.46.0](#), [GPGME-1.23.2](#) (with Python bindings), [OpenSSH-9.8p1](#) (runtime, to access ssh://... repositories), [Pygments-2.18.0](#), [rustc-1.80.1](#) (see `rust/README.rst` and `rust/rhg/README.md`), [Subversion-1.14.3](#) (with Python bindings), [Bazaar](#), [CVS](#), [pyflakes](#), [pyOpenSSL](#), and [re2](#)

Installation of Mercurial

Build Mercurial by issuing the following command:

```
make build
```

To build the documentation (requires [docutils-0.21.2](#)), issue:

```
make doc
```

If you wish to run the tests, the rust tests must be removed as they are currently broken due to syntax errors. To do this, issue:

```
sed -i '138,142d' Makefile
```

To run the test suite, issue:

```
TESTFLAGS="-j<N> --tmpdir tmp" make check
```

where `<N>` is an integer between one and the number of (processor X threads), inclusive. Tests may fail because some error messages have changed in Python or some deprecation warnings are printed that were not present when the test was designed. Two tests are known to fail: `test-duplicateoptions.py` and `test-profile.t`.

In order to investigate any apparently failing tests, you may use the `run-tests.py` script. To see the almost forty switches, some of them very useful, issue `tests/run-tests.py --help`. Running the following commands, you will execute only the tests that failed before:

```
pushd tests &&
rm -rf tmp &&
./run-tests.py --tmpdir tmp test-gpg.t
popd
```

Normally, the previous failures will be reproducible. However, if you add the switch `--debug` before `--tmpdir`, and run the tests again, some failures may disappear, which is a problem with the test suite. If this happens, there will be no more of these failures even if you do not pass the `--debug` switch again.

An interesting switch is `--time`, which will generate a table of all the executed tests and their respective start, end, user, system and real times once the tests are complete. Note that these switches may be used with `make check` by including them in the `TESTFLAGS` environment variable.

Install Mercurial by running the following command (as `root`):

```
make PREFIX=/usr install-bin
```

If you built the documentation, install it by running the following command (as `root`):

```
make PREFIX=/usr install-doc
```

After installation, two very quick and simple tests should run correctly. The first one needs some configuration:

```
cat >> ~/.hgrc << "EOF"
[ui]
username = <user_name> <user@mail>
EOF
```

where you must replace `<user_name>` and `<your@mail>` (mail is optional and can be omitted). With the user identity defined, run `hg debuginstall` and several lines will be displayed, the last one reading "no problems detected". Another quick and simple test is just `hg`, which should output basic commands that can be used with `hg`.

Configuring Mercurial

Config Files

`/etc/mercurial/hgrc` and `~/.hgrc`

The great majority of extensions are disabled by default. Run `hg help extensions` if you need to enable any, e.g. when investigating test failures. This will output a list of enabled and disabled extensions, as well as more information such as how to enable or disable extensions using configuration files.

If you have installed [make-ca-1.14](#) and want Mercurial to use the certificates, as the `root` user, issue:

```
install -v -d -m755 /etc/mercurial &&
cat > /etc/mercurial/hgrc << "EOF"
[web]
cacerts = /etc/pki/tls/certs/ca-bundle.crt
EOF
```

Contents

Installed Programs: `hg`

Installed Libraries: several internal modules under `/usr/lib/python3.12/site-packages/mercurial`

Installed Directories: `/etc/mercurial`, `/usr/lib/python3.12/site-packages/hgemandimport`, `/usr/lib/python3.12/site-packages/hgext`, `/usr/lib/python3.12/site-packages/hgext3rd`, `/usr/lib/python3.12/site-packages/mercurial`, and `/usr/lib/python3.12/site-packages/mercurial-6.8.1-py3.12.egg-info`

Short Descriptions

`hg` is the mercurial version control system

Introduction to NASM

NASM (Netwide Assembler) is an 80x86 assembler designed for portability and modularity. It includes a disassembler as well.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.nasm.us/pub/nasm/releasebuilds/2.16.03/nasm-2.16.03.tar.xz>
- Download MD5 sum: 2b8c72c52eee4f20085065e68ac83b55
- Download size: 1008.1 KB
- Estimated disk space required: 41 MB
- Estimated build time: 0.2 SBU

Additional Downloads

- Optional documentation: <https://www.nasm.us/pub/nasm/releasebuilds/2.16.03/nasm-2.16.03-xdoc.tar.xz>

NASM Dependencies

Optional (for generating documentation):

[asciidoc-10.2.1](#) and [xmlto-0.0.29](#)

Installation of NASM

If you downloaded the optional documentation, put it into the source tree:

```
tar -xf ../nasm-2.16.03-xdoc.tar.xz --strip-components=1
```

Install NASM by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you downloaded the optional documentation, install it with the following instructions as the `root` user:

```
install -m755 -d      /usr/share/doc/nasm-2.16.03/html  &&  
cp -v doc/html/*.html /usr/share/doc/nasm-2.16.03/html  &&  
cp -v doc/*.{txt,ps,pdf} /usr/share/doc/nasm-2.16.03
```

Contents

Installed Programs: nasm and ndisasm

Installed Libraries: None

Installed Directory: /usr/share/doc/nasm-2.16.03

Short Descriptions

`nasm` is a portable 80x86 assembler
`ndisasm` is an 80x86 binary file disassembler

Patchelf-0.18.0

Introduction to Patchelf

The patchelf package contains a small utility to modify the dynamic linker and RPATH of ELF executables.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/NixOS/patchelf/releases/download/0.18.0/patchelf-0.18.0.tar.gz>
- Download MD5 sum: b02099b0d63f06b3fe370f4edfc0c085
- Download size: 448 KB
- Estimated disk space required: 20 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Installation of Patchelf

Install patchelf by running the following commands:

```
./configure --prefix=/usr \
            --docdir=/usr/share/doc/patchelf-0.18.0 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: patchelf

Short Descriptions

`patchelf` is a simple utility for modifying existing ELF executables and libraries; it can change the dynamic loader ("ELF interpreter") of executables and change the RPATH of executables and libraries.

Perl Modules

Introduction to Perl Modules

The Perl module packages (also referred to as Distributions, because each can contain multiple modules) add useful objects to the Perl language. The packages listed on this page are required or recommended for other packages in the book. If they have dependent modules, those are either on this page or else on the next page ([Perl Module Dependencies](#)).

In many cases, only the required or recommended dependencies are listed - there might be other modules which allow more tests to be run, but omitting them will still allow the tests to `PASS`.

For a few modules, the BLFS editors have determined that other modules still listed as prerequisites are not required, and omitted them.

Where an alphabetically-earlier dependency of the same module pulls in a dependency, it is not mentioned for the later dependencies of the same module. You should build the listed dependencies in order.

It is generally worth running the tests for perl modules, they often can show problems such as missing dependencies which are required to use the module. Here, the editors have attempted to separate those dependencies which are only required for running test suites, but they will not be mentioned for a module where one of its dependencies uses that module for its own test suite. That is to say, if you intend to run the test suites, please run them for each dependency of the module.

It is possible to automatically install the current versions of a module and *all* missing or too-old dependencies *recommended by upstream* using CPAN. See [CPAN automated install of perl modules](#) at the end of this page.

Most of these modules only install files below `/usr/lib/perl5/site_perl/5.xx.y` and those will not be documented. One or two install programs (mostly, perl scripts), or a library, into `/usr/bin/` or `/usr/lib/` and these are documented.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/perl-modules>

- [Archive-Zip-1.68](#)
- [autovivification-0.18](#)
- [Business-ISBN-3.009](#)
- [Business-ISMN-1.204](#)

- [Business-ISSN-1.005](#)
- [Class-Accessor-0.51](#)
- [Data-Compare-1.29](#)
- [Data-Dump-1.25](#)
- [Data-Uniqid-0.12](#)
- [DateTime-Calendar-Julian-0.107](#)
- [DateTime-Format-Builder-0.83](#)
- [Encode-EUCJPASCII-0.03](#)
- [Encode-HanExtra-0.23](#)
- [Encode-JIS2K-0.05](#)
- [File-FcntlLock-0.22](#)
- [File-Slurper-0.014](#)
- [File-Which-1.27](#)
- [HTML-Parser-3.83](#)
- [HTTP-Daemon-6.16](#)
- [IO-Socket-SSL-2.088](#)
- [IO-String-1.08](#)
- [IPC-Run3-0.049](#)
- [libwww-perl-6.77](#)
- [Lingua-Translit-0.29](#)
- [List-AllUtils-0.19](#)
- [List-MoreUtils-0.430](#)
- [Log-Log4perl-1.57](#)
- [LWP-Protocol-https-6.14](#)
- [Module-Build-0.4234](#)
- [Net-DNS-1.46](#)
- [Parse-RecDescent-1.967015](#)
- [Parse-Yapp-1.21](#)
- [PerlIO-utf8_strict-0.010](#)
- [Regexp-Common-2024080801](#)
- [SGMLSpm-1.1](#)
- [Sort-Key-1.33](#)
- [Test-Command-0.11](#)
- [Test-Differences-0.71](#)
- [Text-BibTeX-0.89](#)
- [Text-CSV-2.04](#)
- [Text-Roman-3.5](#)
- [Unicode-Collate-1.31](#)
- [Unicode-LineBreak-2019.001](#)
- [URI-5.28](#)
- [XML-LibXML-Simple-1.01](#)
- [XML-LibXSLT-2.003000](#)
- [XML-Simple-2.25](#)
- [XML-Writer-0.900](#)
- [CPAN automated install of perl modules](#)

Archive::Zip-1.68

Introduction to Archive::Zip

The Archive::Zip module allows a Perl program to create, manipulate, read, and write Zip archive files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/P/PH/PHRED/Archive-Zip-1.68.tar.gz>
- Download MD5 sum: a33993309322164867c99e04a4000ee3

Archive::Zip Dependencies

Recommended (for the test suite)

[UnZip-6.0](#) (with its patch)

Installation of Archive::Zip

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: crc32

Short Descriptions

`crc32` computes and prints to stdout the CRC-32 values of the given files

autovivification-0.18

Introduction to the autovivification module

This module allows you disable autovivification (the automatic creation and population of new arrays and hashes whenever undefined variables are dereferenced), and optionally throw a warning or an error when it would have occurred.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/V/VP/VPIT/autovivification-0.18.tar.gz>
- Download MD5 sum: 8dec994e1e7d368e055f21a5777385a0

Installation of autovivification

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Business::ISBN-3.009

Introduction to Business::ISBN

The Business::ISBN module provides functions for working with International Standard Book Numbers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/B/BR/BRIANDFOY/Business-ISBN-3.009.tar.gz>
- Download MD5 sum: e906867846f4d1fa57366aa519da5846

Business::ISBN Dependencies

Required

[Business-ISBN-Data-20240807.001](#)

Installation of Business::ISBN

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Business::ISMN-1.204

Introduction to Business::ISMN

The Business::ISMN module provides functions for working with International Standard Music Numbers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/B/BR/BRIANDFOY/Business-ISMN-1.204.tar.gz>
- Download MD5 sum: dbc9023703262ee29d1b9e9a8294106c

Business::ISMN Dependencies

Required

[Tie-Cycle-1.228](#)

Installation of Business::ISMN

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Business::ISSN-1.005

Introduction to Business::ISSN

The Business::ISSN module provides functions for working with International Standard Serial Numbers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/B/BD/BDF0Y/Business-ISSN-1.005.tar.gz>
- Download MD5 sum: f46bf5585d6c3aa9fb32127edb13151a

Installation of Business::ISSN

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Class::Accessor-0.51

Introduction to Class::Accessor

Class::Accessor generates accessors/mutators for your class.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/K/KA/KASEI/Class-Accessor-0.51.tar.gz>
- Download MD5 sum: 1f1e5990f87cad7659b292fed7dc0407

Installation of Class::Accessor

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Data::Compare-1.29

Introduction to Data::Compare

The Data::Compare module compares two perl data structures.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/D/DC/DCANTRELL/Data-Compare-1.29.tar.gz>
- Download MD5 sum: ce9cb42ba6af634f5ab51f13f37e2ddb

Data::Compare Dependencies

Required

[Clone-0.46](#) and [File-Find-Rule-0.34](#)

Installation of Data::Compare

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Data::Dump-1.25

Introduction to Data::Dump

Data::Dump provides pretty printing of data structures.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/G/GA/GARU/Data-Dump-1.25.tar.gz>
- Download MD5 sum: 9bd7131ef0441e1e0e001bf85e9fae31

Installation of Data::Dump

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Data::Uniqid-0.12

Introduction to Data::Uniqid

Data::Uniqid provides three simple routines for generating unique IDs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/M/MW/MWX/Data-Uniqid-0.12.tar.gz>
- Download MD5 sum: 6bab3b5da09fedfdf60ce2629a7367db

Installation of Data::Uniqid

Although the final test fails and reports an Error, the test suite returns a status of 0. The error can safely be ignored.

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

DateTime::Calendar::Julian-0.107

Introduction to DateTime::Calendar::Julian

DateTime::Calendar::Julian implements the Julian Calendar.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/W/WY/WYANT/DateTime-Calendar-Julian-0.107.tar.gz>
- Download MD5 sum: abd775d1d82f0f45d4fd6214cf7bbcd8

DateTime::Calendar::Julian Dependencies

Required

[DateTime-1.65](#)

Installation of DateTime::Calendar::Julian

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

DateTime::Format::Builder-0.83

Introduction to DateTime::Format::Builder

DateTime::Format::Builder creates DateTime parser classes and objects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/D/DR/DROLSKY/DateTime-Format-Builder-0.83.tar.gz>
- Download MD5 sum: aa41917ca9ad69b3898728ce9c2fb477

DateTime::Format::Builder Dependencies

Required

[DateTime-Format-Srtptime-1.79](#) and [Params-Validate-1.31](#)

Installation of DateTime::Format::Builder

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Encode::EUCJPASCII-0.03

Introduction to Encode::EUCJPASCII

Encode::EUCJPASCII provides an eucJP-open (Extended Unix Code, Japanese) mapping.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/N/NE/NEZUMI/Encode-EUCJPASCII-0.03.tar.gz>
- Download MD5 sum: 5daa65f55b7c2050bb0713d9e95f239d

Installation of Encode::EUCJPASCII

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Encode::HanExtra-0.23

Introduction to Encode::HanExtra

The `Encode::HanExtra` module provides extra sets of Chinese Encodings which are not included in the core `Encode` module because of size issues.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/A/AU/AUDREYT/Encode-HanExtra-0.23.tar.gz>
- Download MD5 sum: e1d3bc32c1c8ee304235a06fbcd5d5a4

Installation of Encode::HanExtra

This module uses the 'unsafe' build and installation instructions (In perl-5.26.0 the use of the current directory in `@INC` was removed for security reasons and this module has not been updated) :

```
PERL_USE_UNSAFE_INC=1 perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Encode::JIS2K-0.05

Introduction to Encode::JIS2K

The `Encode::JIS2K` module provides JIS X 0212 (aka JIS 2000) Encodings.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/D/DA/DANKOGAI/Encode-JIS2K-0.05.tar.gz>
- Download MD5 sum: 06acd9e878d41ffc354258e265db2875

Installation of Encode::JIS2K

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::FcntlLock-0.22

Introduction to File::FcntlLock

File::FcntlLock is a module to do file locking in an object oriented fashion using the [fcntl\(2\)](#) system call. This allows locks on parts of a file as well as on the whole file and overcomes some known problems with [flock\(2\)](#), on which which Perl's `flock()` function is based per default.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/J/JT/JTT/File-FcntlLock-0.22.tar.gz>
- Download MD5 sum: 579698d735d864ee403674f1175f789d

Installation of File::FcntlLock

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::Slurper-0.014

Introduction to File::Slurper

File::Slurper is a simple, sane and efficient module to slurp a file.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/L/LE/LEONT/File-Slurper-0.014.tar.gz>
- Download MD5 sum: d43bc5f069035eff3b6b7c418b4cedc4

File::Slurper Dependencies

Recommended (required for the test suite)

[Test-Warnings-0.033](#)

Installation of File::Slurper

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::Which-1.27

Introduction to File::Which

File::Which provides a portable implementation of the 'which' utility.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/P/PL/PLICEASE/File-Which-1.27.tar.gz>
- Download MD5 sum: d5c9154262b93398f0750ec364207639

Installation of File::Which

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTML::Parser-3.83

Introduction to HTML::Parser

The HTML::Parser distribution is a collection of modules that parse and extract information from HTML documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/O/OA/OALDERS/HTML-Parser-3.83.tar.gz>
- Download MD5 sum: 17a4c886024bfad1cffcbe6b46cda128

HTML::Parser Dependencies

Required

[HTML-Tagset-3.24](#) and [HTTP-Message-6.46](#) (strictly speaking, not required for building, but its module HTTP::Headers is required for tests and a run-time requirement for HTML::HeadParser from this distribution).

Installation of HTML::Parser

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTTP::Daemon-6.16

Introduction to HTTP::Daemon

Instances of the HTTP::Daemon class are HTTP/1.1 servers that listen on a socket for incoming requests. The HTTP::Daemon is a subclass of IO::Socket::INET, so you can perform socket operations directly on it too.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/O/OA/OALDERS/HTTP-Daemon-6.16.tar.gz>
- Download MD5 sum: 51425462790165aeafc2819a7359706f

HTTP::Daemon Dependencies

Required

[HTTP-Message-6.46](#)

Note

Makefile.PL and running the tests will complain that Module::Build::Tiny is missing, but that is only needed for Build.PL which is used if the system lacks a C compiler.

Installation of HTTP::Daemon

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

IO::Socket::SSL-2.088

Introduction to IO::Socket::SSL

IO::Socket::SSL makes using SSL/TLS much easier by wrapping the necessary functionality into the familiar IO::Socket interface and providing secure defaults whenever possible.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/S/SU/SULLR/IO-Socket-SSL-2.088.tar.gz>
- Download MD5 sum: 3e95ca11fdfd5db9cfc0d318ecb6836f

IO::Socket::SSL Dependencies

Required

[make-ca-1.14](#) and [Net-SSLeay-1.94](#)

Recommended

[URI-5.28](#) (to access international domain names)

Installation of IO::Socket::SSL

This module uses a variant of the standard build and installation instructions:

```
yes | perl Makefile.PL &&
make             &&
make test
```

One test, Client non-SSL connection, is known to fail.

Now, as the `root` user:

```
make install
```

Command Explanations

`yes`: Perl will ask if you wish to run external tests, which will 'fail soft' if there are network problems. The default is 'y', which will allow you to script the build.

IO::String-1.08

Introduction to IO::String

IO::String - Emulate file interface for in-core strings.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/G/GA/GAAS/IO-String-1.08.tar.gz>
- Download MD5 sum: 250e5424f290299fc3d6b5d1e9da3835

Installation of IO::String

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

IPC::Run3-0.049

Introduction to IPC::Run3

IPC::Run3 is used to run a subprocess with input/output redirection.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/R/RJ/RJBS/IPC-Run3-0.049.tar.gz>
- Download MD5 sum: 569393ca56dfb78dedd79e93c0439159

Installation of IPC::Run3

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Lingua::Translit-0.29

Introduction to Lingua::Translit

Lingua::Translit and its `translit` program transliterate text between writing systems.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/A/AL/ALINKE/Lingua-Translit-0.29.tar.gz>
- Download MD5 sum: 605a82f06b05fef4fc18bf069b1be511

Installation of Lingua::Translit

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: translit

Short Descriptions

`translit` transliterates text between writing systems using various standards

LWP (libwww-perl-6.77)

Introduction to LWP - The World-wide Web library for Perl

The libwww-perl collection is a set of Perl modules which provides a simple and consistent application programming interface (API) to the World-Wide Web. The main focus of the library is to provide classes and functions that allow you to write WWW clients. The library also contains modules that are of more general use and even classes that help you implement simple HTTP servers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/O/OA/OALDERS/libwww-perl-6.77.tar.gz>
- Download MD5 sum: e70e2f41a97b8d97608569d10b75b931

libwww-perl Dependencies

Required

[File-Listing-6.16](#), [HTTP-CookieJar-0.014](#), [HTTP-Cookies-6.11](#), [HTTP-Daemon-6.16](#), [HTTP-Negotiate-6.01](#), [HTML-Parser-3.83](#), [Net-HTTP-6.23](#), [Try-Tiny-0.31](#) and [WWW-RobotRules-6.02](#)

Recommended (required for the test suite)

[Test-Fatal-0.017](#), [Test-Needs-0.002010](#) and [Test-RequiresInternet-0.05](#)

Installation of libwww-perl

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

After installing this package, if you want HTTPS protocol support install [LWP-Protocol-https-6.14](#).

Contents

Installed Programs: lwp-download, lwp-dump, lwp-mirror, lwp-request

Short Descriptions

lwp-download	is a script to fetch a large file from the web
lwp-dump	is used to see what headers and content is returned for a URL
lwp-mirror	is a simple mirror utility
lwp-request	is a simple command line user agent

List::AllUtils-0.19

Introduction to List::AllUtils

The List::Allutils module combines List::Util and List::MoreUtils in one bite-sized package.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/D/DR/DROLSKY/List-AllUtils-0.19.tar.gz>
- Download MD5 sum: 86469b1f6819ba181a8471eb932965f2

List::AllUtils Dependencies

Required

[List-SomeUtils-0.59](#) and [List-UtilsBy-0.12](#)

Installation of List::AllUtils

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

List::MoreUtils-0.430

Introduction to List::MoreUtils

List::MoreUtils provides the stuff missing in List::Util.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/R/RE/REHSACK/List-MoreUtils-0.430.tar.gz>
- Download MD5 sum: daccd6310021231b827dcc943ff1c6b7

List::MoreUtils Dependencies

Required

[Exporter-Tiny-1.006002](#) and [List-MoreUtils-XS-0.430](#)

Installation of List::MoreUtils

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Log::Log4perl-1.57

Introduction to Log::Log4perl

Log::Log4perl provides a Log4j implementation for perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/E/ET/ETJ/Log-Log4perl-1.57.tar.gz>
- Download MD5 sum: acbe29cbaf03f4478a13579a275b0011

Installation of Log::Log4perl

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: l4p-temp1

Short Descriptions

`l4p-temp1` prints out the text of a template Log4perl configuration for starting a new Log4perl configuration file

LWP::Protocol::https-6.14

Introduction to LWP::Protocol::https

LWP::Protocol::https provides https support for LWP::UserAgent (i.e. [libwww-perl-6.77](#)). Once the module is installed LWP is able to access sites using HTTP over SSL/TLS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/O/OA/OALDERS/LWP-Protocol-https-6.14.tar.gz>
- Download MD5 sum: 65401e3e34be653c9e3b31f798ed5454

LWP::Protocol::https Dependencies

Required

[IO-Socket-SSL-2.088](#), [libwww-perl-6.77](#), and [make-ca-1.14](#) with /etc/pki/tls/certs/ca-bundle.crt.

Installation of LWP::Protocol::https

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

Module::Build-0.4234

Introduction to Module::Build

Module::Build allows perl modules to be built without a `make` command being present.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/L/LE/LEONT/Module-Build-0.4234.tar.gz>
- Download MD5 sum: 0032d0c0bc36a3b68ef41c947829d5e3

Installation of Module::Build

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

Note that this module can also be built using `Build.PL`

Contents

Installed Programs: config_data

Short Descriptions

`config_data` is used to query or change the configuration of perl modules

Net::DNS-1.46

Introduction to Net::DNS

Net::DNS is a DNS resolver implemented in Perl. It can be used to perform nearly any type of DNS query from a Perl script.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/N/NL/NLNETLABS/Net-DNS-1.46.tar.gz>
- Download MD5 sum: 640f572abed308ca5399d3429cbce6bd

Installation of Net::DNS

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Parse::RecDescent-1.967015

Introduction to Parse::RecDescent

Parse::RecDescent incrementally generates top-down recursive-descent text parsers from simple yacc-like grammar specifications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/J/JT/JTBRAUN/Parse-RecDescent-1.967015.tar.gz>
- Download MD5 sum: 7a36d45d62a9b68603edcdbc276006cc

Installation of Parse::RecDescent

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Parse::Yapp-1.21

Introduction to Parse::Yapp

Parse::Yapp is a Perl extension for generating and using LALR parsers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/W/WB/WBRASWELL/Parse-Yapp-1.21.tar.gz>
- Download MD5 sum: 69584d5b0f0304bb2a23cffcd982c5de

Installation of Parse::Yapp

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: yapp

Short Descriptions

`yapp` is a frontend to the Parse::Yapp module, which lets you create a Perl OO parser from an input grammar file

PerlIO::utf8_strict-0.010

Introduction to PerlIO::utf8_strict

PerlIO::utf8_strict provides a fast and correct UTF-8 PerlIO layer. Unlike Perl's default :utf8 layer it checks the input for correctness.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://www.cpan.org/authors/id/L/LE/LEONT/PerlIO-utf8_strict-0.010.tar.gz
- Download MD5 sum: d90ca967f66e05ad9221c79060868346

PerlIO::utf8_strict Dependencies

Recommended (required for the test suite)

[Test-Exception-0.43](#)

Installation of PerlIO::utf8_strict

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Regexp::Common-2024080801

Introduction to Regexp::Common

Regexp::Common provides commonly requested regular expressions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/A/AB/ABIGAIL/Regexp-Common-2024080801.tar.gz>
- Download MD5 sum: 73d4b4b2a0690f9ab573d54a69c22aee

Installation of Regexp::Common

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

SGMLSpm-1.1

Introduction to SGMLSpm

The SGMLSpm module is a Perl library used for parsing the output from James Clark's SGMLS and NSGMLS parsers. This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/R/RA/RAAB/SGMLSpm-1.1.tar.gz>
- Download MD5 sum: 746c74ae969992cedb1a2879b4168090

Installation of SGMLSpm

Before beginning the build, issue the following command to prevent an error:

```
chmod -v 644 MYMETA.yml
```

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

After the package has been installed, run the following command as the `root` user:

```
ln -sv sgmlspl.pl /usr/bin/sgmlspl
```

Contents

Installed Programs: sgmlspl.pl, sgmlspl

Short Descriptions

`sgmlspl.pl` is an SGML processor
`sgmlspl` is a symbolic link used during the install of [DocBook-utils-0.6.14](#)

Sort::Key-1.33

Introduction to Sort::Key

Sort::Key provides a set of functions to sort lists of values by some calculated key value.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/S/SA/SALVA/Sort-Key-1.33.tar.gz>
- Download MD5 sum: a37ab0da0cfdc26e57b4c79e39f6d98f

Installation of Sort::Key

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Command-0.11

Introduction to Test::Command

Test::Command tests the exit status, STDOUT, or STDERR, of an external command.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/D/DA/DANBOO/Test-Command-0.11.tar.gz>
- Download MD5 sum: 9ab83c4695961dbe92cd86efe08f0634

Installation of Test::Command

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Differences-0.71

Introduction to Test::Differences

Test::Differences tests strings and data structures and shows the differences if they do not match.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/D/DC/DCANTRELL/Test-Differences-0.71.tar.gz>
- Download MD5 sum: 23a54d5ada6ffe0850f42cd768b4b3c1

Test::Differences Dependencies

Required

[Text-Diff-1.45](#)

Recommended (required for the test suite)

[Capture-Tiny-0.48](#)

Installation of Test::Differences

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Text::BibTeX-0.89

Introduction to Text::BibTeX

Text::BibTeX provides an interface to read and parse BibTeX files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/A/AM/AMBS/Text-BibTeX-0.89.tar.gz>
- Download MD5 sum: 67fa6d9c03c659627d9fd58bb380c1fe

Text::BibTeX Dependencies

Required

[Config-AutoConf-0.320](#) and [ExtUtils-LibBuilder-0.08](#)

Installation of Text::BibTeX

This module is built using `Build.PL`:

```
perl Build.PL &&
./Build &&
./Build test
```

Now, as the `root` user:

```
./Build install
```

Contents

Installed Programs: `biblex`, `libparse`, `dumpnames`

Installed Libraries: `libbtparse.so`

Short Descriptions

<code>biblex</code>	performs lexical analysis on a BibTeX file
<code>libparse</code>	parses a series of BibTeX files with command line options to control the string post-processing behavior
<code>dumpnames</code>	parses a BibTeX file, splitting 'author' and 'editor' fields into lists of names, and then dumps everything to stdout
<code>libbtparse.so</code>	is a library for parsing and processing BibTeX data files

Text::CSV-2.04

Introduction to Text::CSV

Text::CSV is a comma-separated values manipulator, using XS (eXternal Subroutine - for subroutines written in C or C++) or pure perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/I/IS/ISHIGAKI/Text-CSV-2.04.tar.gz>
- Download MD5 sum: 500b1cd1349e2186877c39aa404230dc

Text::CSV Dependencies

Recommended

[Text-CSV_XS-1.56](#) (required by [biber-2.20](#))

Installation of Text::CSV

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Text::Roman-3.5

Introduction to Text::Roman

Text::Roman allows conversion between Roman and Arabic algorithms (number systems, e.g. MCMXLV and 1945).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/S/SY/SYP/Text-Roman-3.5.tar.gz>
- Download MD5 sum: 1f6b09c0cc1f4425b565ff787a39fd83

Installation of Text::Roman

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Unicode::Collate-1.31

Introduction to Unicode::Collate

Unicode::Collate provides a Unicode collation algorithm.

Note

This is a core module. If you are using perl-5.28.0 or later, its version is good enough for [biber-2.20](#) and you do not need to reinstall this module.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/S/SA/SADAHIRO/Unicode-Collate-1.31.tar.gz>
- Download MD5 sum: ee4d960d057c5e5b02ebb49d0286db8f

Installation of Unicode::Collate

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Unicode::LineBreak-2019.001

Introduction to Unicode::LineBreak

Unicode::LineBreak provides a UAX #14 Unicode Line Breaking Algorithm.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/N/NE/NEZUMI/Unicode-LineBreak-2019.001.tar.gz>
- Download MD5 sum: 003d6da7a13700e069afed9238c864b9

Unicode::LineBreak Dependencies

Required

[MIME-Charset-1.013.1](#) and [Wget-1.24.5](#) (to download two files from unicode.org in the test suite)

Optional

[libthai](#) (to break Thai words into segments)

Installation of Unicode::LineBreak

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

URI-5.28

Introduction to URI

This module implements the URI class. Objects of this class represent "Uniform Resource Identifier references" as specified in RFC 2396 (and updated by RFC 2732). A Uniform Resource Identifier is a compact string of characters that identifies an abstract or physical resource. A Uniform Resource Identifier can be further classified as either a Uniform Resource Locator (URL) or a Uniform Resource Name (URN). The distinction between URL and URN does not matter to the URI class interface. A "URI-reference" is a URI that may have additional information attached in the form of a fragment identifier.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/O/OA/OALDERS/URI-5.28.tar.gz>
- Download MD5 sum: 82aaecd8861f31444338231801ba01ca

URI Dependencies

Recommended (required for the test suite)

[Test-Fatal-0.017](#), [Test-Needs-0.002010](#), and [Test-Warnings-0.033](#)

Optional

[Business-ISBN-3.009](#)

Installation of URI

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

XML::LibXML::Simple-1.01

Introduction to XML::LibXML::Simple

The XML::LibXML::Simple module is a rewrite of XML::Simple to use the XML::LibXML parser for XML structures, instead of the plain Perl or SAX parsers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/M/MA/MARKOV/XML-LibXML-Simple-1.01.tar.gz>
- Download MD5 sum: faad5ed26cd83998f6514be199c56c38

XML::LibXML::Simple Dependencies

Required

[XML-LibXML-2.0210](#)

Installation of XML::LibXML::Simple

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

XML::LibXSLT-2.003000

Introduction to XML::LibXSLT

XML-LibXSLT provides an interface to [libxslt-1.1.42](#)

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/S/SH/SHLOMIF/XML-LibXSLT-2.003000.tar.gz>
- Download MD5 sum: 632dce587b3c405edd4e622364750191

XML::LibXSLT Dependencies

Required

[libxslt-1.1.42](#) and [XML-LibXML-2.0210](#)

Installation of XML::LibXSLT

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

XML::Simple-2.25

Introduction to XML::Simple

XML::Simple provides an easy API to read and write XML (especially config files). It is deprecated and its use is discouraged.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/G/GR/GRANTM/XML-Simple-2.25.tar.gz>
- Download MD5 sum: bb841dce889a26c89a1c2739970e9fbc

XML::Simple Dependencies

Optional

[XML-SAX-1.02](#) (for an alternative parser which will be used if available, otherwise XML::Parser (which was installed in LFS) will be used)

Installation of XML::Simple

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

XML::Writer-0.900

Introduction to XML::Writer

XML::Writer provides a Perl extension for writing XML documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cpan.org/authors/id/J/JO/JOSEPHW/XML-Writer-0.900.tar.gz>
- Download MD5 sum: 2457214360cefda445742a608dd6195e

Installation of XML::Writer

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
```

```
make test
```

Now, as the `root` user:

```
make install
```

CPAN automated install of perl modules

Automatic Installation of Perl Modules.

There is an alternate way of installing the modules using the `cpan` shell `install` command. The command automatically downloads the latest source from the CPAN archive for the module and any missing prerequisite modules listed by upstream. Then for each module it extracts it, runs the compilation, the tests and installs it.

You still need to install any non-perl dependencies before running the automated installation method. You may wish to clean out the `build/` directory after installing, to free up the space. If any post-install actions such as creating a symlink are mentioned, you should also do those.

The first time you run `cpan`, you'll be prompted to enter some information regarding download locations and methods. This information is retained in files located in `~/.cpan`.

In particular, you may wish to configure it so that [Sudo-1.9.15p5](#) is used for the installs, allowing you to build and test as a regular user. The following examples have not used that approach.

Start the `cpan` shell by issuing '`cpan`' as the `root` user. Any module may now be installed from the `cpan>` prompt with the command:

```
install <Module::Name>
```

For additional commands and help, issue '`help`' from the `cpan>` prompt.

Alternatively, for scripted or non-interactive installations, use the following syntax as the `root` user to install one or more modules:

```
cpan -i <Module1::Name> <Module2::Name>
```

Review the `cpan.1` man page for additional parameters you can pass to `cpan` on the command line.

Perl Module Dependencies

Perl Modules which are only required by other modules

The modules on the previous page are referenced from other pages in BLFS, but these modules are only in the book as dependencies of those modules. If you use the CPAN install method, you do not need to read this page.

The BLFS editors pay much less attention to these modules, and the versions will not be regularly reviewed. In all cases, only the required or recommended dependencies are listed - there might be other modules which allow more tests to be run, but omitting them will still allow the tests to `PASS`.

Note

The links on this page (to metacpan.org) should go to "known good" versions, for which their dependencies are correct. If you wish to use a later version, please check the Changes file at <https://metacpan.org> - sometimes added dependencies are listed, other times not. Some of these modules have very frequent updates, often bringing different dependencies. The linked metacpan.org versions below were known to work with the module versions in BLFS when last reviewed.

However, if you notice that the Changes file for a newer version than is in the current development book reports a fix for a security issue, please report this to either the `blfs-dev` or the `blfs-support` list.

Similarly, if you discover that an updated module on the previous page needs an extra dependency, please report this.

- [Algorithm-Diff-1.201](#)
- [Alien-Build-2.83](#)
- [Alien-Build-Plugin-Download-GitLab-0.01](#)

- [Alien-Libxml2-0.19](#)
- [B-COW-0.007](#)
- [B-Hooks-EndOfScope-0.28](#)
- [Business-ISBN-Data-20240807.001](#)
- [Capture-Tiny-0.48](#)
- [Class-Data-Inheritable-0.09](#)
- [Class-Inspector-1.36](#)
- [Class-Singleton-1.6](#)
- [Class-Tiny-1.008](#)
- [Clone-0.46](#)
- [Config-AutoConf-0.320](#)
- [CPAN-Meta-Check-0.018](#)
- [DateTime-1.65](#)
- [DateTime-Format-Srptime-1.79](#)
- [DateTime-Locale-1.43](#)
- [DateTime-TimeZone-2.62](#)
- [Devel-StackTrace-2.05](#)
- [Dist-CheckConflicts-0.11](#)
- [Encode-Locale-1.05](#)
- [Eval-Closure-0.14](#)
- [Exception-Class-1.45](#)
- [Exporter-Tiny-1.006002](#)
- [ExtUtils-LibBuilder-0.08](#)
- [FFI-CheckLib-0.31](#)
- [File-chdir-0.1011](#)
- [File-Copy-Recursive-0.45](#)
- [File-Find-Rule-0.34](#)
- [File-Listing-6.16](#)
- [File-ShareDir-1.118](#)
- [File-ShareDir-Install-0.14](#)
- [HTML-Tagset-3.24](#)
- [HTTP-CookieJar-0.014](#)
- [HTTP-Cookies-6.11](#)
- [HTTP-Date-6.06](#)
- [HTTP-Message-6.46](#)
- [HTTP-Negotiate-6.01](#)
- [IO-HTML-1.004](#)
- [IPC-System-Simple-1.30](#)
- [List-MoreUtils-XS-0.430](#)
- [List-SomeUtils-0.59](#)
- [List-SomeUtils-XS-0.58](#)
- [List-UtilsBy-0.12](#)
- [LWP-MediaTypes-6.04](#)
- [MIME-Charset-1.013.1](#)
- [Module-Implementation-0.09](#)
- [Module-Runtime-0.016](#)
- [MRO-Compat-0.15](#)
- [namespace-autoclean-0.29](#)
- [namespace-clean-0.27](#)
- [Net-HTTP-6.23](#)
- [Net-SSLeay-1.94](#)

- [Number-Compare-0.03](#)
- [Package-Stash-0.40](#)
- [Params-Validate-1.31](#)
- [Params-ValidationCompiler-0.31](#)
- [Path-Tiny-0.146](#)
- [Role-Tiny-2.002004](#)
- [Scope-Guard-0.21](#)
- [Specio-0.48](#)
- [Sub-Exporter-Progressive-0.001013](#)
- [Sub-Identify-0.14](#)
- [Sub-Quote-2.006008](#)
- [Sub-Uplevel-0.2800](#)
- [Term-Table-0.022](#)
- [Test-Deep-1.204](#)
- [Test-Exception-0.43](#)
- [Test-Fatal-0.017](#)
- [Test-File-1.993](#)
- [Test-File-ShareDir-1.001002](#)
- [Test-LeakTrace-0.17](#)
- [Test-Needs-0.002010](#)
- [Test-Requires-0.11](#)
- [Test-RequiresInternet-0.05](#)
- [Test-Simple-1.302200](#)
- [Test-utf8-1.02](#)
- [Test-Warnings-0.033](#)
- [Test-Without-Module-0.23](#)
- [Test2-Plugin-NoWarnings-0.10](#)
- [Text-CSV XS-1.56](#)
- [Text-Diff-1.45](#)
- [Text-Glob-0.11](#)
- [Tie-Cycle-1.228](#)
- [TimeDate-2.33](#)
- [Try-Tiny-0.31](#)
- [Variable-Magic-0.64](#)
- [WWW-RobotRules-6.02](#)
- [XML-LibXML-2.0210](#)
- [XML-NamespaceSupport-1.12](#)
- [XML-SAX-1.02](#)
- [XML-SAX-Base-1.09](#)

Algorithm::Diff-1.201

Introduction to Algorithm::Diff

Algorithm::Diff computes 'intelligent' differences between two files or lists.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RJ/RJBS//Algorithm-Diff-1.201.tar.gz>
- Download MD5 sum: 2eaae910f5220261ee2bbdfc4a8df2c2

Installation of Algorithm::Diff

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Alien::Build-2.83

Introduction to Alien::Build

Alien::Build provides tools for building external (non-CPAN) dependencies for CPAN.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PL/PLICEASE/Alien-Build-2.83.tar.gz>
- Download MD5 sum: fbec7ad3281181db0cd3e2219710815e

Alien::Build Dependencies

Required

[Capture-Tiny-0.48](#), [File-Which-1.27](#), [FFI-CheckLib-0.31](#) and [File-chdir-0.1011](#)

Installation of Alien::Build

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Alien::Build::Plugin::Download::GitLab-0.01

Introduction to Alien::Build::Plugin::Download::GitLab

Alien::Build::Plugin::Download::GitLab allows Alien::Build to download from GitLab (in practice, this does not download if the required library is already installed on the system).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PL/PLICEASE/Alien-Build-Plugin-Download-GitLab-0.01.tar.gz>
- Download MD5 sum: ad1d815262ad7dd98b0a9b35ba2f05ef

Alien::Build::Plugin::Download::GitLab Dependencies

Required

[Alien-Build-2.83](#) and [URI-5.28](#)

Installation of Alien::Build::Plugin::Download::GitLab

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Alien::Libxml2-0.19

Introduction to Alien::Libxml2

Alien::Libxml2 is designed to allow modules to install the C libxml2 library on your system. In BLFS, it uses `pkg-config` to find how to link to the installed [libxml2-2.13.3](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PL/PLICEASE/Alien-Libxml2-0.19.tar.gz>
- Download MD5 sum: 54ef82ddf7641279a72f216e405f9a5e

Alien::Libxml2 Dependencies

Required

[Alien-Build-Plugin-Download-GitLab-0.01](#), [libxml2-2.13.3](#), and [Path-Tiny-0.146](#)

Recommended (required for the test suite)

[Test-Simple-1.302200](#)

Installation of Alien::Libxml2

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

B::COW-0.007

Introduction to B::COW

B::COW provides additional helpers for the B core module to check Copy On Write.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/A/AT/ATOOMIC/B-COW-0.007.tar.gz>
- Download MD5 sum: 7afc46f19e6f906e2ba5769b21fca5ff

Installation of B::COW

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

B::Hooks::EndOfScope-0.28

Introduction to B::Hooks::EndOfScope

`B::Hooks::EndOfScope` allows you to execute code when Perl finishes compiling the surrounding scope.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/ET/ETHER/B-Hooks-EndOfScope-0.28.tar.gz>
- Download MD5 sum: d738ba65539d4acd601d47cc3e2cbb3a

B::Hooks::EndOfScope Dependencies

Required

[Module-Implementation-0.09](#), [Sub-Exporter-Progressive-0.001013](#) and [Variable-Magic-0.64](#)

Recommended (required for the test suite)

[Try-Tiny-0.31](#)

Installation of B::Hooks::EndOfScope

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Business::ISBN::Data-20240807.001

Introduction to Business-ISBN-Data

`Business-ISBN-Data` is a data pack for `Business::ISBN`.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/B/BR/BRIANDFOY/Business-ISBN-Data-20240807.001.tar.gz>
- Download MD5 sum: 814d1eccb79435ff67ac44556da18d5

Installation of Business-ISBN-Data

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
```

```
make test
```

Now, as the `root` user:

```
make install
```

Capture::Tiny-0.48

Introduction to Capture::Tiny

The Capture::Tiny module captures STDOUT and STDERR from Perl, XS (eXternal Subroutine, i.e. written in C or C++) or external programs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/Capture-Tiny-0.48.tar.gz>
- Download MD5 sum: f5d24083ad270f8326dd659dd83eeb54

Installation of Capture::Tiny

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Class::Data::Inheritable-0.09

Introduction to Class::Data::Inheritable

Class::Data::Inheritable is for creating accessor/mutators to class data. That is, if you want to store something about your class as a whole (instead of about a single object).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RS/RSHERER/Class-Data-Inheritable-0.09.tar.gz>
- Download MD5 sum: bd25ecd6e5d528fbc3783edf1b8facef

Installation of Class::Data::Inheritable

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Class::Inspector-1.36

Introduction to Class::Inspector

Class::Inspector allows you to get information about a loaded class.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PL/PLICEASE/Class-Inspector-1.36.tar.gz>
- Download MD5 sum: 084c3aec023639d21ecbaf7d4460b21

Installation of Class::Inspector

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Class::Singleton-1.6

Introduction to Class::Singleton

A Singleton describes an object class that can have only one instance in any system, such as a print spooler. This module implements a Singleton class from which other classes can be derived.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/S/SH/SHAY/Class-Singleton-1.6.tar.gz>
- Download MD5 sum: d9c84a7b8d1c490c38e88ed1f9faae47

Installation of Class::Singleton

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Class::Tiny-1.008

Introduction to Class::Tiny

Class::Tiny offers a minimalist class construction kit.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/Class-Tiny-1.008.tar.gz>
- Download MD5 sum: e3ccfae5f64d443e7e1110be964d7202

Installation of Class::Tiny

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Clone-0.46

Introduction to Clone

Clone recursively copies perl datatypes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/G/GA/GARU/Clone-0.46.tar.gz>
- Download MD5 sum: cafa8984a2c2e005e54b27dd1e3f0afe

Clone Dependencies

Recommended (required for the test suite)

[B-COW-0.007](#)

Installation of Clone

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Config::AutoConf-0.320

Introduction to Config::AutoConf

The Config::AutoConf module implements some of the AutoConf macros (detecting a command, detecting a library, etc.) in pure perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/A/AM/AMBS/Config-AutoConf-0.320.tar.gz>
- Download MD5 sum: 71664b2864232e265179ac29298e0916

Config::AutoConf Dependencies

Required

[Capture-Tiny-0.48](#) and [File-Slurper-0.014](#)

Installation of Config::AutoConf

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

CPAN::Meta::Check-0.018

Introduction to CPAN::Meta::Check

CPAN::Meta::Check verifies if requirements described in a CPAN::Meta object are present.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/L/LE/LEONT/CPAN-Meta-Check-0.018.tar.gz>
- Download MD5 sum: d1c2190e8bc1c176b9ee9cba3ac403ad

Installation of CPAN::Meta::Check

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

DateTime-1.65

Introduction to DateTime

DateTime is a date and time object for perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/DateTime-1.65.tar.gz>
- Download MD5 sum: 6f60018500f8f20c5fd3d34495eae1eb

DateTime Dependencies

Required

[DateTime-Locale-1.43](#) and [DateTime-TimeZone-2.62](#)

Recommended (required for the test suite)

[CPAN-Meta-Check-0.018](#), [Test-Fatal-0.017](#), [Test-Warnings-0.033](#) and [Test-Without-Module-0.23](#)

Installation of DateTime

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

DateTime::Format::Strptime-1.79

Introduction to DateTime::Format::Strptime

DateTime::Format::Strptime implements most of [strftime\(3\)](#), i.e. it takes a string and a pattern and returns a DateTime object.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/DateTime-Format-Strptime-1.79.tar.gz>
- Download MD5 sum: 441cfec62b0b8a1b4c05cbe5ef73fbf4

DateTime::Format::Strptime Dependencies

Required

[DateTime-1.65](#)

Installation of DateTime::Format::Strptime

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

DateTime::Locale-1.43

Introduction to DateTime::Locale

DateTime::Locale provides localization support for [DateTime-1.65](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/DateTime-Locale-1.43.tar.gz>
- Download MD5 sum: 24b801b592354fe58e67040fdb1cc00c

DateTime::Locale Dependencies

Required

[Dist-CheckConflicts-0.11](#), [File-ShareDir-1.118](#), [namespace-autoclean-0.29](#) and [Params-ValidationCompiler-0.31](#)

Recommended (required for the test suite)

[CPAN-Meta-Check-0.018](#), [IPC-System-Simple-1.30](#) and [Test-File-ShareDir-1.001002](#)

Installation of DateTime::Locale

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

DateTime::TimeZone-2.62

Introduction to DateTime::TimeZone

This class is the base class for all time zone objects. A time zone is represented internally as a set of observances, each of which describes the offset from GMT for a given time period.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/DateTime-TimeZone-2.62.tar.gz>
- Download MD5 sum: e11bb3a07c4f3d156f7dc356816f3962

DateTime::TimeZone Dependencies

Required

[Class-Singleton-1.6](#), [Module-Runtime-0.016](#), and [Params-ValidationCompiler-0.31](#)

Recommended (required for the test suite)

Both [Test-Fatal-0.017](#) and [Test-Requires-0.11](#), but only if a copy of [DateTime-1.65](#) (for which this is a dependency) has already been installed

Installation of DateTime::TimeZone

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Devel::StackTrace-2.05

Introduction to Devel::StackTrace

Devel::StackTrace provides an object representing a stack trace.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Devel-StackTrace-2.05.tar.gz>
- Download MD5 sum: b8ca19bb4c76e98a04373618db9c7c3c

Installation of Devel::StackTrace

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Dist::CheckConflicts-0.11

Introduction to Dist::CheckConflicts

Dist::CheckConflicts declares version conflicts for a distribution, to support post-install updates of dependent distributions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DO/DOY/Dist-CheckConflicts-0.11.tar.gz>
- Download MD5 sum: c8725a92b9169708b0f63036812070f2

Dist::CheckConflicts Dependencies

Required

[Module-Runtime-0.016](#)

Recommended (required for the test suite)

[Test-Fatal-0.017](#)

Installation of Dist::CheckConflicts

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Encode::Locale-1.05

Introduction to Encode::Locale

Encode::Locale determines the locale encoding.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/G/GA/GAAS/Encode-Locale-1.05.tar.gz>
- Download MD5 sum: fcfdb8e4ee34bcf62aed429b4a23db27

Installation of Encode::Locale

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Eval::Closure-0.14

Introduction to Eval::Closure

Eval::Closure safely and cleanly creates closures via string eval.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DO/DOY/Eval-Closure-0.14.tar.gz>
- Download MD5 sum: ceeb1fc579ac9af981fa6b600538c285

Eval::Closure Dependencies

Recommended (required for the test suite)

[Test-Fatal-0.017](#) and [Test-Requires-0.11](#)

Installation of Eval::Closure

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Exception::Class-1.45

Introduction to Exception::Class

Exception::Class allows you to declare real exception classes in Perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Exception-Class-1.45.tar.gz>
- Download MD5 sum: 1e564d20b374a99fdf660ba3f36b0098

Exception::Class Dependencies

Required

[Class-Data-Inheritable-0.09](#) and [Devel-StackTrace-2.05](#)

Installation of Exception::Class

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Exporter::Tiny-1.006002

Introduction to Exporter::Tiny

Exporter::Tiny is an exporter with the features of Sub::Exporter but only core dependencies.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/T/TO/TOBYINK/Exporter-Tiny-1.006002.tar.gz>
- Download MD5 sum: 0545ee8f4edcb9dc5a87b21ed25edd74

Installation of Exporter::Tiny

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

ExtUtils::LibBuilder-0.08

Introduction to ExtUtils::LibBuilder

ExtUtils::LibBuilder is a tool to build C libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/A/AM/AMBS/ExtUtils-LibBuilder-0.08.tar.gz>
- Download MD5 sum: 8ffe9e9a3c2f916f40dc4f6aed237d33

ExtUtils::LibBuilder Dependencies

Required

[Module-Build-0.4234](#)

Installation of ExtUtils::LibBuilder

This module is built using `Build.PL`:

```
perl Build.PL &&
./Build &&
./Build test
```

Now, as the `root` user:

```
./Build install
```

FFI::CheckLib-0.31

Introduction to FFI::CheckLib

FFI::CheckLib checks whether a particular dynamic library is available for FFI (Foreign Function Interface) to use.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PL/PLICEASE/FFI-CheckLib-0.31.tar.gz>
- Download MD5 sum: ffc8e61bb686dd631bed3ddf102af41c

FFI::CheckLib Dependencies

Recommended (required for the test suite)

[Capture-Tiny-0.48](#), [File-Which-1.27](#), [Path-Tiny-0.146](#), and [Test-Simple-1.302200](#)

Installation of FFI::CheckLib

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::chdir-0.1011

Introduction to File::chdir

File::chdir provides a more sensible way to change directories.

Perl's `chdir()` has the unfortunate problem of being very, very, very global. If any part of your program calls `chdir()` or if any library you use calls `chdir()`, it changes the current working directory for the *whole* program. File::chdir gives you an alternative, `$CWD` and `@CWD`.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/File-chdir-0.1011.tar.gz>
- Download MD5 sum: 932090f6c5f602301ae66c259de23ebb

Installation of File::chdir

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::Copy::Recursive-0.45

Introduction to File::Copy::Recursive

This module copies and moves directories recursively (or single files), to an optional depth and attempts to preserve each file or directory's mode.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DM/DMUEY/File-Copy-Recursive-0.45.tar.gz>
- Download MD5 sum: e5eee1a3f8ae3aebbac063ea54870e54

File::Copy::Recursive Dependencies

Recommended (required for the test suite)

[Path-Tiny-0.146](#), [Test-Deep-1.204](#), [Test-Fatal-0.017](#), [Test-File-1.993](#), and [Test-Warnings-0.033](#)

Installation of File::Copy::Recursive

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::Find::Rule-0.34

Introduction to File::Find::Rule

File::Find::Rule is a friendlier interface to File::Find. It allows you to build rules which specify the desired files and directories.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RC/RCLAMP/File-Find-Rule-0.34.tar.gz>
- Download MD5 sum: a7aa9ad4d8ee87b2a77b8e3722768712

File::Find::Rule Dependencies

Required

[Number-Compare-0.03](#) and [Text-Glob-0.11](#)

Installation of File::Find::Rule

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: findrule

Short Descriptions

`findrule` is a command line wrapper to File::Find::Rule

File::Listing-6.16

Introduction to File::Listing

File::Listing parses a directory listing.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PL/PLICEASE/File-Listing-6.16.tar.gz>
- Download MD5 sum: d4fc8b0c86633d1fa5bf75323720eadc

File::Listing Dependencies

Required

[HTTP-Date-6.06](#)

Installation of File::Listing

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::ShareDir-1.118

Introduction to File::ShareDir

File::ShareDir allows you to access data files which have been installed by File::ShareDir::Install.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RE/REHSACK/File-ShareDir-1.118.tar.gz>
- Download MD5 sum: 0084f730f4e3d4d89703d92b3ea82f54

File::ShareDir Dependencies

Required

[Class-Inspector-1.36](#) and [File-ShareDir-Install-0.14](#)

Installation of File::ShareDir

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

File::ShareDir::Install-0.14

Introduction to File::ShareDir::Install

`File::ShareDir::Install` allows you to install read-only data files from a distribution.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): [https://cpan.metacpan.org/authors/id/E/ET/ETHER/File-S...tar.gz](https://cpan.metacpan.org/authors/id/E/ET/ETHER/File-S...)
- Download MD5 sum: bac4d924f3d863b00648ab56ec0dcdbc

Installation of `File::ShareDir::Install`

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTML::Tagset-3.24

Introduction to `HTML::Tagset`

`HTML::Tagset` provides several data tables useful in parsing HTML.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): [https://cpan.metacpan.org/authors/id/P/PE/PETDANCE/HTML-Tagset-3.24tar.gz](https://cpan.metacpan.org/authors/id/P/PE/PETDANCE/HTML-Tagset-3.24.tar.gz)
- Download MD5 sum: f8db8974f5e7fe7df2a58263a7b00552

Installation of `HTML::Tagset`

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTTP::CookieJar-0.014

Introduction to `HTTP::CookieJar`

`HTTP::CookieJar` provides a minimalist HTTP user agent cookie jar.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): [https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/HTTP-CookieJar-0.014tar.gz](https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/HTTP-CookieJar-0.014.tar.gz)
- Download MD5 sum: a1d891ce0046f1a2c19e2c617d624d0d

HTTP::CookieJar Dependencies

Required

[HTTP-Date-6.06](#)

Recommended (required for the test suite)

[Test-Deep-1.204](#), [Test-Requires-0.11](#), and [URI-5.28](#)

Installation of HTTP::CookieJar

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTTP::Cookies-6.11

Introduction to HTTP::Cookies

HTTP::Cookies provides a class for objects that represent a "cookie jar" -- that is, a database of all the HTTP cookies that a given LWP::UserAgent (from [libwww-perl-6.77](#)) object knows about.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/O/OA/OALDERS/HTTP-Cookies-6.11.tar.gz>
- Download MD5 sum: 80017e7e56bdc8ba16dea75789748829

HTTP::Cookies Dependencies

Required

[HTTP-Message-6.46](#)

Installation of HTTP::Cookies

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTTP::Date-6.06

Introduction to HTTP::Date

HTTP::Date provides functions to deal with the date formats used by the HTTP protocol and also with some other date formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/O/OA/OALDERS/HTTP-Date-6.06.tar.gz>
- Download MD5 sum: 60462359bfeb1e6d14602508cf07885

HTTP::Date Dependencies

Recommended

[TimeDate-2.33](#) (to allow it to recognize zones other than GMT and numeric)

Installation of HTTP::Date

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTTP::Message-6.46

Introduction to HTTP::Message

HTTP::Message provides a base class for HTTP style message objects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/O/OA/OALDERS/HTTP-Message-6.46.tar.gz>
- Download MD5 sum: 12a4bf7d993ba7b231df9a24f8bf3ec5

HTTP::Message Dependencies

Required

[Clone-0.46](#), [Encode-Locale-1.05](#), [HTTP-Date-6.06](#), [IO-HTML-1.004](#), [LWP-MediaTypes-6.04](#), and [URI-5.28](#)

Recommended (required for the test suite)

[Test-Needs-0.002010](#) and [Try-Tiny-0.31](#)

Installation of HTTP::Message

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

HTTP::Negotiate-6.01

Introduction to HTTP::Negotiate

HTTP::Negotiate provides a complete implementation of the HTTP content negotiation algorithm.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/G/GA/GAAS/HTTP-Negotiate-6.01.tar.gz>
- Download MD5 sum: 1236195250e264d7436e7bb02031671b

HTTP::Negotiate Dependencies

Recommended (required for the test suite)

[HTTP-Message-6.46](#)

Installation of HTTP::Negotiate

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

IO::HTML-1.004

Introduction to IO::HTML

IO::HTML opens an HTML file with automatic character set detection.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/C/CJ/CJM/IO-HTML-1.004.tar.gz>
- Download MD5 sum: 04bbe363686fd19bfb4cc0ed775e3d03

Installation of IO::HTML

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

IPC::System::Simple-1.30

Introduction to IPC::System::Simple

IPC::System::Simple takes the hard work out of calling external commands and producing detailed diagnostics.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/J/JK/JKEENAN/IPC-System-Simple-1.30.tar.gz>
- Download MD5 sum: e68341fd958fd013b3521d909904f675

Installation of IPC::System::Simple

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

List::MoreUtils::XS-0.430

Introduction to List::MoreUtils::XS

List::MoreUtils::XS is a compiled backend for List::MoreUtils

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RE/REHSACK/List-MoreUtils-XS-0.430.tar.gz>
- Download MD5 sum: e77113e55b046906aecfb4ddb4f0c662

Installation of List::MoreUtils::XS

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

List::SomeUtils-0.59

Introduction to List::SomeUtils

List::SomeUtils provides the stuff missing in List::Util.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/List-SomeUtils-0.59.tar.gz>
- Download MD5 sum: 333b4adb2907deff2be8da5899881453

List::SomeUtils Dependencies

Required

[Module-Implementation-0.09](#) and [List-SomeUtils-XS-0.58](#)

Installation of List::SomeUtils

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

List::SomeUtils::XS-0.58

Introduction to List::SomeUtils::XS

List::SomeUtils::XS is a (faster) XS (eXternal Subroutine) implementation for List::SomeUtils.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/List-SomeUtils-XS-0.58.tar.gz>
- Download MD5 sum: 396eabe83a75fc8d7542d95812469d1

List::SomeUtils::XS Dependencies

Recommended (required for the test suite)

[Test-LeakTrace-0.17](#) and [Test-Warnings-0.033](#)

Installation of List::SomeUtils::XS

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

List::UtilsBy-0.12

Introduction to List::UtilsBy

List::UtilsBy provides a number of higher-order list utility functions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PE/PEVANS/List-UtilsBy-0.12.tar.gz>
- Download MD5 sum: 54a8c7092bc02f29ea6c5ae215eea385

Installation of List::UtilsBy

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

LWP::MediaTypes-6.04

Introduction to LWP::MediaTypes

LWP::MediaTypes guesses the media type (i.e. the MIME Type) for a file or URL.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/O/OA/OALDERS/LWP-MediaTypes-6.04.tar.gz>
- Download MD5 sum: 84b799a90c0d2ce52897a7cb4c0478d0

LWP::MediaTypes Dependencies

Recommended (required for the test suite)

[Test-Fatal-0.017](#)

Installation of LWP::MediaTypes

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make           &&
make test
```

Now, as the `root` user:

```
make install
```

MIME::Charset-1.013.1

Introduction to MIME::Charset

MIME::Charset provides information about character sets used for MIME messages on the internet, such as their encodings.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/N/NE/NEZUMI/MIME-Charset-1.013.1.tar.gz>
- Download MD5 sum: b1932fcf806c8deb1b4a20d6afbfa8ac

MIME::Charset Dependencies

Recommended

[Encode-EUCJPASCII-0.03](#), [Encode-HanExtra-0.23](#) and [Encode-JIS2K-0.05](#) (because all are required by [biber-2.20](#))

Installation of MIME::Charset

This module uses a variant of the standard build and installation instructions:

```
yes '' | perl Makefile.PL &&
make           &&
make test
```

Now, as the `root` user:

```
make install
```

Command Explanations

`yes 1`: Perl will ask if you wish to install a further module for handling legacy Japanese encodings, and another which it would use for translating documentation to Japanese. The default option is 'n', using 'yes' allows you to script the build.

Module::Implementation-0.09

Introduction to Module::Implementation

Module::Implementation loads one of several alternate underlying implementations of a module (e.g. eXternal Subroutine or pure Perl, or an implementation for a given OS).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Module-Implementation-0.09.tar.gz>
- Download MD5 sum: 52e3fe0ca6b1eff0488d59b7aacc0667

Module::Implementation Dependencies

Required

[Module-Runtime-0.016](#) and [Try-Tiny-0.31](#)

Recommended (required for the test suite)

[Test-Fatal-0.017](#) and [Test-Requires-0.11](#)

Installation of Module::Implementation

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Module::Runtime-0.016

Introduction to Module::Runtime

Module::Runtime provides functions to deal with runtime handling of Perl modules.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/Z/ZE/ZEFRAM/Module-Runtime-0.016.tar.gz>
- Download MD5 sum: d3d47222fa2e3dfcb4526f6cc8437b20

Installation of Module::Runtime

Although Module::Build is still listed as a prerequisite, it is no-longer necessary on systems with a working `make`.

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

MRO::Compat-0.15

Introduction to MRO::Compat

The "mro" namespace provides several utilities for dealing with method resolution order and method caching in general in Perl 5.9.5 and higher. This module provides those interfaces for earlier versions of Perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/H/HA/HAARG/MRO-Compat-0.15.tar.gz>
- Download MD5 sum: f644dafe901214cedfa7ed8b43b56df1

Installation of MRO::Compat

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

namespace::autoclean-0.29

Introduction to namespace::autoclean

This module is very similar to `namespace::clean`, except it will clean all imported functions, no matter if you imported them before or after you used the pragma. It will also not touch anything that looks like a method.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/ET/ETHER/namespace-autoclean-0.29.tar.gz>
- Download MD5 sum: 39b38c776cd1f0ee03cc70781a2f2798

namespace::autoclean Dependencies

Required

[namespace-clean-0.27](#) and [Sub-Identify-0.14](#)

Recommended (required for the test suite)

[Test-Needs-0.002010](#)

Installation of namespace::autoclean

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

namespace::clean-0.27

Introduction to namespace::clean

This package allows you to keep imports and functions out of your namespace.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RI/RIBASUSHI/namespace-clean-0.27.tar.gz>
- Download MD5 sum: cba97f39ef7e594bd8489b4fdcd662

namespace::clean Dependencies

Required

[B-Hooks-EndOfScope-0.28](#) and [Package-Stash-0.40](#)

Installation of namespace::clean

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Net::HTTP-6.23

Introduction to Net::HTTP

The Net::HTTP class is a low level HTTP client. An instance of the class represents a connection to an HTTP server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/O/OA/OALDERS/Net-HTTP-6.23.tar.gz>
- Download MD5 sum: 1682735ddd1c059864ca5c1bbf15ab95

Net::HTTP Dependencies

Required

[URI-5.28](#)

Installation of Net::HTTP

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Net::SSLeay-1.94

Introduction to Net::SSLeay

Net::SSLeay is a Perl extension for using OpenSSL.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/C/CH/CHRISN/Net-SSLeay-1.94.tar.gz>

- Download MD5 sum: 1b22c764e5a094c6261e37a4b1f148ce

Installation of Net::SSLeay

Note

If enabling the external tests, one test in t/external/15_altnames.t may fail.

This module uses a variant of the standard build and installation instructions:

```
yes '' | perl Makefile.PL &&
make          &&
make test
```

Now, as the `root` user:

```
make install
```

Command Explanations

`yes ''`: Perl will ask if you wish to run external tests, which will fail if you do not have network connectivity. The default is '`n`', specifying this allows you to script the build.

Number::Compare-0.03

Introduction to Number::Compare

Number::Compare compiles a simple comparison to an anonymous subroutine, which you can call with a value to be tested against. It understands IEC standard magnitudes (k, ki, m, mi, g, gi).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RC/RCLAMP/Number-Compare-0.03.tar.gz>
- Download MD5 sum: ded4085a8fc96328742785574ca65208

Installation of Number::Compare

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make          &&
make test
```

Now, as the `root` user:

```
make install
```

Package::Stash-0.40

Introduction to Package::Stash

Manipulating stashes (Perl's symbol tables) is occasionally necessary, but incredibly messy, and easy to get wrong. This module hides all of that behind a simple API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/ET/ETHER/Package-Stash-0.40.tar.gz>
- Download MD5 sum: 7a2922941cc2aad6a52642e4fb13d07b

Package::Stash Dependencies

Required

[Dist-CheckConflicts-0.11](#) and [Module-Implementation-0.09](#)

Recommended (required for the test suite)

[CPAN-Meta-Check-0.018](#), [Test-Fatal-0.017](#), and [Test-Needs-0.002010](#)

Installation of Package::Stash

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Params::Validate-1.31

Introduction to Params::Validate

Params::Validate allows you to validate method or function call parameters to an arbitrary level of specificity.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Params-Validate-1.31.tar.gz>
- Download MD5 sum: ef5f57387c2c9032b59fb23023cf5b25

Params::Validate Dependencies

Required

[Module-Build-0.4234](#) and [Module-Implementation-0.09](#)

Recommended (required for the test suite)

[Test-Fatal-0.017](#) and [Test-Requires-0.11](#)

Installation of Params::Validate

This module is built using `Build.PL`:

```
perl Build.PL &&
./Build &&
./Build test
```

Now, as the `root` user:

```
./Build install
```

Params::ValidationCompiler-0.31

Introduction to Params::ValidationCompiler

Params::ValidationCompiler builds an optimized subroutine parameter validator.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Params-ValidationCompiler-0.31.tar.gz>
- Download MD5 sum: 15528055f3f53c8cfbbe1f928dec07

Params::ValidationCompiler Dependencies

Required

[Exception-Class-1.45](#) and [Specio-0.48](#)

Recommended (required for the test suite)

[Test-Without-Module-0.23](#) and [Test2-Plugin-NoWarnings-0.10](#)

Installation of Params::ValidationCompiler

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Path::Tiny-0.146

Introduction to Path::Tiny

Path::Tiny provides a small fast utility for working with file paths.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/Path-Tiny-0.146.tar.gz>
- Download MD5 sum: 656d3556bb7f30c77d0881d564e200b2

Installation of Path::Tiny

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Role::Tiny-2.002004

Introduction to Role::Tiny

Role::Tiny is a minimalist role composition tool.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/H/HA/HAARG/Role-Tiny-2.002004.tar.gz>
- Download MD5 sum: 9ee45591befa3d0b1094ac75d282b6ba

Installation of Role::Tiny

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Scope::Guard-0.21

Introduction to Scope::Guard

Scope::Guard provides a convenient way to perform cleanup or other forms of resource management at the end of a scope.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/C/CH/CHOCOLATE/Scope-Guard-0.21.tar.gz>
- Download MD5 sum: be57b915d23ddac7677ef2ad9e52b92a

Installation of Scope::Guard

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Specio-0.48

Introduction to Specio

Specio provides classes for representing type constraints and coercion, along with syntax sugar for declaring them.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Specio-0.48.tar.gz>
- Download MD5 sum: 96cf1ae4e2e205986e03672071116b16

Specio Dependencies

Required

[Devel-StackTrace-2.05](#), [Eval-Closure-0.14](#), [Module-Runtime-0.016](#), [Role-Tiny-2.002004](#), [Sub-Quote-2.006008](#), and [Try-Tiny-0.31](#)

Recommended (required for the test suite)

Optional

[namespace-autoclean-0.29](#) (for the test suite)

Installation of Specio

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Sub::Exporter::Progressive-0.001013

Introduction to Sub::Exporter::Progressive

Sub::Exporter::Progressive is a wrapper for Sub::Exporter.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/F/FR/FREW/Sub-Exporter-Progressive-0.001013.tar.gz>
- Download MD5 sum: 72cf6acdd2a0a8b105821a4db98e4ebe

Installation of Sub::Exporter::Progressive

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Sub::Identify-0.14

Introduction to Sub::Identify

Sub::Identify allows you to retrieve the real name of code references. It is encouraged to migrate to Sub::Util (a core module) when possible.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RG/RGARCIA/Sub-Identify-0.14.tar.gz>
- Download MD5 sum: 014f19e72698b6a2cbc54adc9691825

Installation of Sub::Identify

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
```

```
make test
```

Now, as the `root` user:

```
make install
```

Sub::Quote-2.006008

Introduction to Sub::Quote

Sub::Quote provides ways to generate subroutines from strings.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/H/HA/HAARG/Sub-Quote-2.006008.tar.gz>
- Download MD5 sum: f19c60039ba87f69f7f9357fc0a03e07

Installation of Sub::Quote

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Sub::Uplevel-0.2800

Introduction to Sub::Uplevel

Sub::Uplevel allows you to fool a caller that it is running in a higher stack frame.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DA/DAGOLDEN/Sub-Uplevel-0.2800.tar.gz>
- Download MD5 sum: 6c6a174861fd160e8d5871a86df00baf

Installation of Sub::Uplevel

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Term::Table-0.022

Introduction to Term::Table

Term::Table formats a header and rows into a table. This is used by some failing tests to provide diagnostics about what has gone wrong.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/EX/EXODIST/Term-Table-0.022.tar.gz>
- Download MD5 sum: 5ec7329895d2c368687b76278afe1d49

Installation of Term::Table

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Deep-1.204

Introduction to Test::Deep

`Test::Deep` gives you very flexible ways to check that the result you got is the result you were expecting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RJ/RJBS/Test-Deep-1.204.tar.gz>
- Download MD5 sum: fcff296434cd92538ae9de9d1744705f

Installation of Test::Deep

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Exception-0.43

Introduction to Test::Exception

`Test::Exception` provides convenience methods for testing exception based code.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/EX/EXODIST/Test-Exception-0.43.tar.gz>
- Download MD5 sum: 572d355026fb0b87fc2b8c64b83cada0

Test::Exception Dependencies

Required

[Sub-Uplevel-0.2800](#)

Installation of Test::Exception

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Fatal-0.017

Introduction to Test::Fatal

The `Test::Fatal` module provides simple helpers for testing code which throws exceptions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RJ/RJBS/Test-Fatal-0.017.tar.gz>
- Download MD5 sum: 2f160c31e1848536e3b82112d573bb76

Test::Fatal Dependencies

Required

[Try-Tiny-0.31](#)

Installation of Test::Fatal

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::File-1.993

Introduction to Test::File

`Test::File` provides a collection of test utilities for file attributes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/B/BD/BDFOY/Test-File-1.993.tar.gz>
- Download MD5 sum: dccb988191187261ce3a2a10af939625

Test::File Dependencies

Recommended (required for the test suite)

[Test-utf8-1.02](#)

Installation of Test::File

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::File::ShareDir-1.001002

Introduction to Test::File::ShareDir

`Test::File::ShareDir` is some low level plumbing to enable a distribution to perform tests while consuming its own share directories in a manner similar to how they will be once installed. This allows [File-ShareDir-1.118](#) to see the latest version of content instead of whatever is installed on the target system where you are testing.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/K/KE/KENTNL/Test-File-ShareDir-1.001002.tar.gz>
- Download MD5 sum: ec31466aa44c1cd56c6cb51d7ec3a5de

Test::File::ShareDir Dependencies

Required

[Class-Tiny-1.008](#), [File-Copy-Recursive-0.45](#), [File-ShareDir-1.118](#), [Path-Tiny-0.146](#), and [Scope-Guard-0.21](#)

Recommended (required for the test suite)

[Test-Fatal-0.017](#)

Installation of Test::File::ShareDir

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::LeakTrace-0.17

Introduction to Test::LeakTrace

`Test::LeakTrace` traces memory leaks.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/L/LE/LEEJO/Test-LeakTrace-0.17.tar.gz>
- Download MD5 sum: afdb2cc6be0807cb635fb601a004d522

Installation of Test::LeakTrace

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Needs-0.002010

Introduction to Test::Needs

Test::Needs skips tests if a requested module is not present.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/H/HA/HAARG/Test-Needs-0.002010.tar.gz>
- Download MD5 sum: 2b3d10946001561297624e7668f09c26

Installation of Test::Needs

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Requires-0.11

Introduction to Test::Requires

The Test::Requires module checks if another (optional) module can be loaded, and if not it skips all the current tests.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/T/TO/TOKUHIROM/Test-Requires-0.11.tar.gz>
- Download MD5 sum: 999d6c4e46ea7baae7a5113292e02ed8

Installation of Test::Requires

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::RequiresInternet-0.05

Introduction to Test::RequiresInternet

Test::RequiresInternet is intended to easily test network connectivity before functional tests begin to connect to non-local Internet resources.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/M/MA/MALLEN/Test-RequiresInternet-0.05.tar.gz>
- Download MD5 sum: 0ba9f1cff4cf90ed2618c2eddf525d8

Installation of Test::RequiresInternet

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Simple-1.302200

Introduction to Test::Simple

Test::Simple contains basic utilities for writing tests.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/EX/EXODIST/Test-Simple-1.302200.tar.gz>
- Download MD5 sum: a954ebd548dc6e4e8b6b82577b2b77fb

Test::Simple Dependencies

Required

[Term-Table-0.022](#)

Installation of Test::Simple

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make             &&
make test
```

Now, as the `root` user:

```
make install
```

Test::utf8-1.02

Introduction to Test::utf8

Test::utf8 is a collection of tests useful for dealing with utf8 strings in Perl.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/M/MA/MARKF/Test-utf8-1.02.tar.gz>
- Download MD5 sum: 71d187539c76ac1ed9a0242ff208796d

Installation of Test::utf8

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Warnings-0.033

Introduction to Test::Warnings

Test::Warnings tests for warnings and the lack of them.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/ET/ETHER/Test-Warnings-0.033.tar.gz>
- Download MD5 sum: a936461688611bd85258c09e4bf6de68

Installation of Test::Warnings

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test::Without::Module-0.23

Introduction to Test::Without::Module

This module allows you to deliberately hide modules from a program even though they are installed. This is mostly useful for testing modules that have a fallback when a certain dependency module is not installed.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/C/CO/CORION/Test-Without-Module-0.23.tar.gz>
- Download MD5 sum: 58a507875131f63a936e0b971dd18f67

Installation of Test::Without::Module

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Test2::Plugin::NoWarnings-0.10

Introduction to Test2::Plugin::NoWarnings

Test2::Plugin::NoWarnings causes tests to fail if there are any warnings while they run.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/D/DR/DROLSKY/Test2-Plugin-NoWarnings-0.10.tar.gz>
- Download MD5 sum: d50e21a76f1fef09004092a73b1c065b

Test2::Plugin::NoWarnings Dependencies

Required

[Test-Simple-1.302200](#)

Recommended (required for the test suite)

[IPC-Run3-0.049](#)

Installation of Test2::Plugin::NoWarnings

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Text::CSV_XS-1.56

Introduction to Text::CSV_XS

Text::CSV_XS provides facilities for the composition and decomposition of comma-separated values.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://cpan.metacpan.org/authors/id/H/HM/HMBRAND/Text-CSV_XS-1.56.tgz
- Download MD5 sum: f945044c171bcd4150570759980280b1

Installation of Text::CSV_XS

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Text::Diff-1.45

Introduction to Text::Diff

`Text::Diff` performs diffs on files and record sets.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/N/NE/NEILB/Text-Diff-1.45.tar.gz>
- Download MD5 sum: edf57b6189f7651a6be454062a4e6d9c

Text::Diff Dependencies

Required

[Algorithm-Diff-1.201](#)

Installation of `Text::Diff`

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Text::Glob-0.11

Introduction to `Text::Glob`

`Text::Glob` implements [glob\(3\)](#) style matching that can be used to match against text, rather than fetching names from a filesystem.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/R/RC/RCLAMP/Text-Glob-0.11.tar.gz>
- Download MD5 sum: d001559c504a2625dd117bd1558f07f7

Installation of `Text::Glob`

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Tie::Cycle-1.228

Introduction to `Tie::Cycle`

You use `Tie::Cycle` to go through a list over and over again. Once you get to the end of the list, you go back to the beginning. You don't have to worry about any of this since the magic of tie does that for you.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/B/BD/BDF0Y/Tie-Cycle-1.228.tar.gz>
- Download MD5 sum: 2c99b09532d80599f4b041f3ecae7af8

Installation of Tie::Cycle

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

TimeDate-2.33

Introduction to TimeDate

TimeDate provides miscellaneous timezone manipulation routines.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/A/AT/ATOOMIC/TimeDate-2.33.tar.gz>
- Download MD5 sum: 5e5afe22c8d417417283d1f7f4572a57

Installation of TimeDate

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Try::Tiny-0.31

Introduction to Try::Tiny

Try::Tiny provides `try` and `catch` to expect and handle exceptional conditions, avoiding quirks in Perl and common mistakes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/E/ET/ETHER/Try-Tiny-0.31.tar.gz>
- Download MD5 sum: 993a29ee8a03c9bd9c2f7c53d1082a03

Try::Tiny Dependencies

Optional (can be used by the test suite)

[Capture-Tiny-0.48](#)

Installation of Try::Tiny

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Variable::Magic-0.64

Introduction to Variable::Magic

Magic is Perl's way of enhancing variables. With this module, you can add your own magic to any variable.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/V/VP/VPIT/Variable-Magic-0.64.tar.gz>
- Download MD5 sum: 957d53fc6614deb593aa6d7cf96d713a

Installation of Variable::Magic

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

WWW::RobotRules-6.02

Introduction to WWW::RobotRules

WWW::RobotRules parses `robots.txt` files, creating a WWW::RobotRules object with methods to check if access to a given URL is prohibited.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/G/GA/GAAS/WWW-RobotRules-6.02.tar.gz>
- Download MD5 sum: b7186e8b8b3701e70c22abf430742403

WWW::RobotRules Dependencies

Required (runtime)

[libwww-perl-6.77](#) (install this module first and install that later)

Installation of WWW::RobotRules

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

XML::LibXML-2.0210

Introduction to XML::LibXML

XML::LibXML is a perl binding for [libxml2-2.13.3](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/S/SH/SHLOMIF/XML-LibXML-2.0210.tar.gz>
- Download MD5 sum: d2bad7f395716a6f57abde538d47008c

XML::LibXML Dependencies

Required

[Alien-Libxml2-0.19](#) and [XML-SAX-1.02](#)

Installation of XML::LibXML

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

XML::NamespaceSupport-1.12

Introduction to XML::NamespaceSupport

XML::NamespaceSupport offers a simple way to process namespaced XML names from within any application that may need them.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/P/PE/PERIGRIN/XML-NamespaceSupport-1.12.tar.gz>
- Download MD5 sum: a8916c6d095bcf073e1108af02e78c97

Installation of XML::NamespaceSupport

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

XML::SAX-1.02

Introduction to XML::SAX

XML::SAX is a SAX parser access API for Perl. It includes classes and APIs required for implementing SAX drivers, along with a factory class for returning any SAX parser installed on the user's system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/G/GR/GRANTM/XML-SAX-1.02.tar.gz>
- Download MD5 sum: b62e3754523695c7f5bbcaca3676a38d

XML::SAX Dependencies

Required

[libxml2-2.13.3](#), [XML-NamespaceSupport-1.12](#), and [XML-SAX-Base-1.09](#)

Installation of XML::SAX

This module uses a variant of the standard build and installation instructions:

```
yes | perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

Command Explanations

`yes`: Perl will ask if you want it to alter `ParserDetails.ini` and then wait for a response. The default is 'y'. Using 'yes' allows you to script this.

XML::SAX::Base-1.09

Introduction to XML::SAX::Base

This module has a very simple task - to be a base class for PerlSAX drivers and filters.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cpan.metacpan.org/authors/id/G/GR/GRANTM/XML-SAX-Base-1.09.tar.gz>
- Download MD5 sum: ec347a14065dd7aec7d9fb181b2d7946

Installation of XML::SAX-Base

This module uses the standard build and installation instructions:

```
perl Makefile.PL &&
make &&
make test
```

Now, as the `root` user:

```
make install
```

PHP-8.3.10

Introduction to PHP

PHP is the PHP Hypertext Preprocessor. Primarily used in dynamic web sites, it allows for programming code to be directly embedded into the HTML markup. It is also useful as a general purpose scripting language.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.php.net/distributions/php-8.3.10.tar.xz>
- Download MD5 sum: 49f47bb78d521ad284ab8f15e91e6c76
- Download size: 12 MB
- Estimated disk space required: 575 MB (with documentation and tests)
- Estimated build time: 1.9 SBU (with parallelism=4; add 1.8 SBU for tests)

Additional Downloads

- Optional pre-built documentation (single file html):
https://www.php.net/distributions/manual/php_manual_en.html.gz
- Optional pre-built documentation (chunked html):
https://www.php.net/distributions/manual/php_manual_en.tar.gz. Note that the documentation can be found in languages other than English at <https://www.php.net/download-docs.php>

PHP Dependencies

Recommended

[Apache-2.4.62](#) and [libxml2-2.13.3](#)

Optional System Utilities and Libraries

[Aspell-0.60.8.1](#), [enchant-2.8.2](#), [libxslt-1.1.42](#), an [MTA](#) (that provides a `sendmail` command), [pcre2-10.44](#), [AppArmor](#), [Dmalloc](#), [Net-SNMP](#), [oniguruma](#), [OSSP mm](#), [re2c](#), and [XMLRPC-EPI](#)

Optional Graphics Utilities and Libraries

[FreeType-2.13.3](#), [libexif-0.6.24](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [a graphical environment](#), [FDF Toolkit](#), [GD](#), and [t1lib](#)

Optional Web Utilities

[cURL-8.9.1](#), [tidy-html5-5.8.0](#), [Caudium](#), [Hyperwave](#), [Roxen WebServer](#), and [WDDX](#)

Optional Data Management Utilities and Libraries

[Imdb-0.9.31](#), [MariaDB-10.11.8](#) or [MySQL](#), [OpenLDAP-2.6.8](#), [PostgreSQL-16.4](#), [SQLite-3.46.1](#), [unixODBC-2.3.12](#), [Berkeley DB](#) (deprecated) [Adabas](#), [Birdstep](#), [cdb](#), [DBMaker](#), [Empress](#), [FrontBase](#), [IBM DB2](#), [libiodbc](#), [Mini SQL](#), [MonetDB](#), and [QDBM](#)

PHP also provides support for many commercial database tools such as Oracle, SAP and ODBC Router.

Optional Security/Encryption Utilities and Libraries

[Cyrus SASL-2.1.28](#), [MIT Kerberos V5-1.21.3](#), [libmcrypt](#), and [mhash](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/php>

Installation of PHP

You can use PHP for server-side scripting, command-line scripting or client-side GUI applications. This book provides instructions for setting up PHP for server-side scripting as it is the most common form.

Note

PHP has many more `configure` options that will enable support for various things. You can use `./configure --help` to see a full list of the available options. Also, use of the [PHP web site](#) is highly recommended, as their online docs are very good. An example of a `configure` command that utilizes many of the most common dependencies can be found at https://anduin.linuxfromscratch.org/BLFS/files/php_configure.txt.

If, for whatever reason, you don't have [libxml2-2.13.3](#) installed, you need to add `--disable-libxml` to the `configure` Command in the instructions below. Note that this will prevent the `pear` command from being built.

Install PHP by running the following commands:

```
./configure --prefix=/usr \
    --sysconfdir=/etc \
    --localstatedir=/var \
    --datadir=/usr/share/php \
    --mandir=/usr/share/man \
    --enable-fpm \
    --without-pear \
    --with-fpm-user=apache \
    --with-fpm-group=apache \
    --with-fpm-systemd \
    --with-config-file-path=/etc \
    --with-zlib \
    --enable-bcmath \
    --with-bz2 \
    --enable-calendar \
    --enable-dba=shared \
    --with-gdbm \
    --with-gmp \
    --enable-ftp \
    --with-gettext \
    --enable-mbstring \
    --disable-mbregex \
    --with-readline \
    && make
```

To test the results, issue: `make test`. Several tests (out of over 18000) may fail, in which case you are asked whether you want to send the report to the PHP developers. If you want to automate the test, you may prefix the command with `yes "n"` | .

Now, as the `root` user:

```
make install \
install -v -m644 php.ini-production /etc/php.ini &&

install -v -m755 -d /usr/share/doc/php-8.3.10 &&
install -v -m644 CODING_STANDARDS* EXTENSIONS NEWS README* UPGRADE* \
/usr/share/doc/php-8.3.10
```

The default configuration files for the fastCGI process manager are installed only if they do not already exist on the system. If this is the first installation, they should be renamed, as the `root` user:

```
if [ -f /etc/php-fpm.conf.default ]; then
  mv -v /etc/php-fpm.conf.default{,.default}
  mv -v /etc/php-fpm.d/www.conf{,.default}
fi
```

The pre-built HTML documentation is packaged in two forms: a tarball containing many individual files, useful for quick loading into your browser, and one large individual file, which is useful for using the search utility of your browser. If you downloaded either, or both, of the documentation files, issue the following commands as the `root` user to install them (note these instructions assume English docs, modify the tarball names below if necessary).

For the "Single HTML" file:

```
install -v -m644 ./php_manual_en.html.gz \
/usr/share/doc/php-8.3.10 &&
gunzip -v /usr/share/doc/php-8.3.10/php_manual_en.html.gz
```

For the "Many HTML files" tarball:

```
tar -xvf ./php_manual_en.tar.gz \
-C /usr/share/doc/php-8.3.10 --no-same-owner
```

The bundled pear is not installed because of a bug which might pollute the filesystem with several hidden files and directories. If pear is needed, execute the following commands to install it:

```
wget https://pear.php.net/go-pear.phar  
php ./go-pear.phar
```

Command Explanations

--datadir=/usr/share/php: This works around a bug in the build machinery, which installs some data to a wrong location.

--enable-fpm: This parameter allows building the fastCGI Process Manager.

--with-fpm-systemd: This parameter allows the FastCGI Process Manager to integrate with systemd.

--without-pear: This switch disables installation of bundled pear software.

--with-config-file-path=/etc: This parameter makes PHP look for the `php.ini` configuration file in `/etc`.

--with-zlib: This parameter adds support for Zlib compression.

--enable-bcmath: Enables bc style precision math functions.

--with-bz2: Adds support for Bzip2 compression functions.

--enable-calendar: This parameter provides support for calendar conversion.

--enable-dba=shared: This parameter enables support for database (dbm-style) abstraction layer functions.

--enable-ftp: This parameter enables FTP functions.

--with-gettext: Enables functions that use Gettext text translation.

--enable-mbstring: This parameter enables multibyte string support.

--with-readline: This parameter enables command line Readline support.

--disable-libxml: This option allows building PHP without libxml2 installed.

--with-apxs2: Instead of building the fastCGI process manager, it is possible to build an apache module. This has some performance penalty for heavy loaded servers, but may be easier to set up. This switch is incompatible with the `--enable-fpm` and `--with-fpm-...` switches.

--with-mysqli=shared: This option includes MySQLi support.

--with-mysql-sock=/run/mysqld/mysqld.sock: Location of the MySQL unix socket pointer.

--with-pdo-mysql=shared: This option includes PDO: MySQL support.

--with-tidy=shared: This option includes tidy library support.

Configuring PHP

Config Files

`/etc/php.ini`, `/etc/pear.conf`, `/etc/php-fpm.conf`, and `/etc/php-fpm.d/www.conf`

Configuration Information

The file used as the default `/etc/php.ini` configuration file is recommended by the PHP development team. This file modifies the default behavior of PHP. If no `/etc/php.ini` is used, all configuration settings fall to the defaults. You should review the comments in this file and ensure the changes are acceptable in your particular environment.

The fastCGI process manager uses the configuration file `/etc/php-fpm.conf`. The default file shipped with PHP includes all the `/etc/php-fpm.d/*.conf` in turn. There is a shipped `/etc/php-fpm.d/www.conf` file, that contains the parameters related to the interaction with the Apache Web server.

You may have noticed the following from the output of the `make install` command:

```
You may want to add: /usr/lib/php to your php.ini include_path
```

If desired, add the entry using the following command as the `root` user:

```
sed -i 's@php/includes"@\ninclude_path = ".:/usr/lib/php"@' \  
/etc/php.ini
```

To enable fastCGI support in the Apache web server, two LoadModule directives must be added to the `httpd.conf` file. They are commented out, so just issue the following command as `root` user:

```
sed -i -e '/proxy_module/s/^#/' \
-e '/proxy_fcgi_module/s/^#/' \
/etc/httpd/httpd.conf
```

Those modules accept various `ProxyPass` directives. One possibility is (as the `root` user):

```
echo \
'ProxyPassMatch ^/(.*\.php)$ fcgi://127.0.0.1:9000/srv/www/$1' >> \
/etc/httpd/httpd.conf
```

Additionally, it may be useful to add an entry for `index.php` to the `DirectoryIndex` directive of the `httpd.conf` file. Lastly, adding a line to set up the `.phps` extension to show highlighted PHP source may be desirable:

```
AddType application/x-httpd-php-source .phps
```

You'll need to restart the Apache web server after making any modifications to the `httpd.conf` file.

Systemd Unit

To start the `php-fpm` daemon at boot, install the `systemd` unit from the [blfs-systemd-units-20240801](#) package by running the following command as the `root` user:

```
make install-php-fpm
```

Contents

Installed Programs: `phar` (symlink), `phar.phar`, `php`, `php-cgi`, `php-config`, `php-fpm`, `phpdbg`, and `phpize`

Installed Libraries: `dba.so` and `opcache.so` in `/usr/lib/php/extensions/no-debug-non-zts-20230831`

Installed Directories: `/etc/php-fpm.d`, `/usr/{include,lib,share}/php`, and `/usr/share/doc/php-8.3.10`

Short Descriptions

<code>php</code>	is a command line interface that enables you to parse and execute PHP code
<code>pear</code>	is the PHP Extension and Application Repository (PEAR) package manager. This isn't installed by default
<code>php-fpm</code>	is the fastCGI process manager for PHP
<code>phpdbg</code>	is the interactive PHP debugger

Python-3.12.5

Introduction to Python 3

The Python 3 package contains the Python development environment. This is useful for object-oriented programming, writing scripts, prototyping large programs or developing entire applications.

Note

Python 3 was installed in LFS. The only reason to rebuild it here is if optional modules are needed, or to upgrade this package.

This package is known to build and work properly using an LFS 12.2 platform.

Important

If upgrading to a new Python-3 minor version (for example, from Python-3.11.x to Python-3.12.0), you will need to reinstall any Python3 modules you have installed. You should also reinstall packages that generate Python3 modules including [GLib-2.80.4](#) (with GObject Introspection), [libxml2-2.13.3](#), [opencv-4.10.0](#), [FontForge-20230101](#), [gnome-tweaks-46.1](#), [Samba-4.20.4](#), and [Graphviz-12.1.0](#) (if swig is installed).

Before you upgrade, you can get a list of modules installed with `pip3 list`. The list may be incomplete as some Python modules are not installed with `pip3`, for example the `cracklib` module installed by [CrackLib-2.10.2](#). Use `ls /usr/lib/python3.minor/site-packages` for a comprehensive list.

The Python modules from LFS will also have to be reinstalled: [flit-core](#), [wheel](#), [setuptools](#), [meson](#), [MarkupSafe](#), and [Jinja2](#).

Package Information

- Download (HTTP): <https://www.python.org/ftp/python/3.12.5/Python-3.12.5.tar.xz>
- Download MD5 sum: 02c7d269e077f4034963bba6befdc715
- Download size: 19 MB
- Estimated disk space required: 335 MB (add 59 MB for tests)
- Estimated build time: 1.6 SBU (add 0.6 SBU for tests; both using parallelism=4)

Additional Optional Download

- Download (HTTP): <https://www.python.org/ftp/python/doc/3.12.5/python-3.12.5-docs-html.tar.bz2>
- Download MD5 sum: 52274d813236ca4a972fb6988480dc56

Python 3 Dependencies

Recommended

[SQLite-3.46.1](#) (required if building firefox or thunderbird)

Optional

[BlueZ-5.77](#), [GDB-15.1](#) (required for some tests), [Valgrind-3.23.0](#), and [libmpdec](#)

Optional (For Additional Modules)

[libnsl-2.0.1](#), [Tk-8.6.14](#), and [Berkeley DB](#) (deprecated)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/Python3>

Installation of Python 3

Install Python 3 by running the following commands:

```
CXX="/usr/bin/g++"          \
./configure --prefix=/usr    \
            --enable-shared \
            --with-system-expat \
            --enable-optimizations &&
make
```

Some tests are known to occasionally hang indefinitely. So to test the results, run the test suite but set a 2-minute time limit for each test case:

```
make test TESTOPTS="--timeout 120"
```

For a relatively slow system you may need to increase the time limit and 1 SBU (measured when building Binutils pass 1 with one CPU core) should be enough. Some tests are flaky, so the test suite will automatically re-run failed tests. If a test failed but then passed when re-run, it should be considered as passed.

Of over 42,000 tests, the following are known to fail: `test_xxsubinterpreters`, `test_import`, `test_interpreters`, and `test_threading`.

Now, as the `root` user:

```
make install
```

If upgrading and the documentation has been downloaded, optionally install it as the `root` user:

```
install -v -dm755 /usr/share/doc/python-3.12.5/html
```

```
tar --strip-components=1 \
--no-same-owner \
--no-same-permissions \
-C /usr/share/doc/python-3.12.5/html \
-xvf ../python-3.12.5-docs-html.tar.bz2
```

Command Explanations

`CXX="/usr/bin/g++" ./configure ...`: Avoid an annoying message during configuration.

`--with-system-expat`: This switch enables linking against the system version of Expat.

`--enable-optimizations`: This switch enables stable, but expensive, optimizations.

`--with-lto`: This optional switch enables thick Link Time Optimization. Unusually, it creates a much larger `/usr/lib/python3.12/config-3.12-<arch>-linux-gnu/libpython3.12.a` with a small increase in the time to compile Python. Runtime results do not appear to show any benefit from doing this.

Configuring Python 3

In order for `python3` to find the installed documentation, create the following version independent symlink:

```
ln -svfn python-3.12.5 /usr/share/doc/python-3
```

and add the following environment variable to the individual user's or system's profile:

```
export PYTHONDOCS=/usr/share/doc/python-3/html
```

Contents

Installed Programs: 2to3 (symlink) and 2to3-3.12, idle3 (symlink) and idle3.12, pip3 and pip3.12, pydoc3 (symlink) and pydoc3.12, python3 (symlink) and python3.12, and python3-config (symlink) and python3.12-config

Installed Libraries: libpython3.12.so and libpython3.so

Installed Directories: /usr/include/python3.12, /usr/lib/python3.12, and /usr/share/doc/python-3.12.5

Short Descriptions

<code>idle3</code>	is a wrapper script that opens a Python aware GUI editor. For this script to run, you must have installed Tk before Python so that the Tkinter Python module is built
<code>pydoc3</code>	is the Python documentation tool
<code>python3</code>	is an interpreted, interactive, object-oriented programming language
<code>python3.12</code>	is a version-specific name for the <code>python</code> program

Python-3.11.1

Introduction to Python 3.11

The Python 3.11 package contains an older version of the Python development environment. This is **only** needed for building [seamonkey-2.53.18.2](#) because its build system has not been updated to support [Python-3.12.5](#).

Note

Do NOT install this package if you are not installing Seamonkey.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.python.org/ftp/python/3.11.1/Python-3.11.1.tar.xz>
- Download MD5 sum: 4efe92adf28875c77d3b9b2e8d3bc44a
- Download size: 19 MB
- Estimated disk space required: 301 MB

- Estimated build time: 1.1 SBU (Using parallelism=4)

Python 3.11 Dependencies

Recommended

[SQLite-3.46.1](#) (required if building firefox or thunderbird)

Installation of Python 3.11

Install Python 3.11 by running the following commands:

```
CXX="/usr/bin/g++" \
./configure --prefix=/opt/python3.11 \
    --disable-shared \
    --with-system-expat \
make
```

Since this package is only used in very limited situations, tests are not recommended.

Now, as the `root` user:

```
make install
```

Command Explanations

`CXX="/usr/bin/g++" ./configure ...`: Avoid an annoying message during configuration.

`--prefix=/opt/python3.11`: This installs python 3.11 into /opt in order to avoid conflicts with the system version of python, and allows for easy removal/isolation once programs update to python3.12 or later.

`--disable-shared`: This switch disables building shared libraries. Since all the packages that need python-3.11 have built in modules and don't directly link to python, it is safe to turn off shared library support.

`--with-system-expat`: This switch enables linking against the system version of Expat.

Contents

Installed Programs: `python3.11`

Installed Libraries: `libpython3.11.a`

Installed Directories: `/opt/python3.11/include/python3.11`, and `/opt/python3.11/lib/python3.11`,

Short Descriptions

`python3.11` is a version-specific name for the `python` program

Python Modules

Introduction to Python Modules

The Python module packages add useful objects to the Python language. Modules utilized by packages throughout BLFS are listed here, along with their dependencies.

Note

If you want to run tests, several of the following modules depend on the `sqlite3` core python module that has not been built in LFS. It is therefore recommended to rebuild Python after installing the recommended dependency listed in [Python-3.12.5](#).

Important

In BLFS, we normally build and install Python 3 modules with `pip3`. Please take care that the `pip3 install` commands in the book should be run as `root` unless it's for a Python virtual environment. Running `pip3 install`

as a non-root user may seem to work fine, but it will cause the installed module to be inaccessible by other users.

`pip3 install` will not reinstall an already installed module by default. For using the `pip3 install` command to upgrade a module (for example, from meson-0.61.3 to meson-0.62.0), insert `--upgrade` into the command line. If it's really necessary to downgrade a module or reinstall the same version for some reason, insert `--force-reinstall` into the command line.

- [asciidoc-10.2.1](#)
- [CacheControl-0.14.0](#)
- [cssselect-1.2.0](#)
- [cython-3.0.11](#)
- [dbusmock-0.32.1](#)
- [D-Bus Python-1.3.2](#)
- [docutils-0.21.2](#)
- [doxypypp-0.8.8.7](#)
- [doxyqml-0.5.3](#)
- [Gi-DocGen-2024.1](#)
- [html5lib-1.1](#)
- [lxml-5.3.0](#)
- [Mako-1.3.5](#)
- [NumPy-2.1.0](#)
- [packaging-24.1](#)
- [ply-3.11](#)
- [psutil-6.0.0](#)
- [Py3c-1.4](#)
- [PyAtSpi2-2.46.1](#)
- [PyCairo-1.26.1](#)
- [pygdbmi-0.11.0.0](#)
- [Pygments-2.18.0](#)
- [PyGObject-3.48.2](#)
- [pyparsing-3.1.2](#)
- [pySerial-3.5](#)
- [pytest-8.3.2](#)
- [PyXDG-0.28](#)
- [PyYAML-6.0.2](#)
- [recommonmark-0.7.1](#)
- [requests-2.32.3](#)
- [sentry-sdk-2.13.0](#)
- [Scour-0.38.2](#)
- [six-1.16.0](#)
- [sphinx-8.0.2](#)
- [sphinx_rtd_theme-2.0.0](#)

Asciidoc-10.2.1

Introduction to Asciidoc Module

The Asciidoc package is a text document format for writing notes, documentation, articles, books, ebooks, slideshows, web pages, man pages and blogs. AsciiDoc files can be translated to many formats including HTML, PDF, EPUB, and man page.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/a/asciidoc/asciidoc-10.2.1.tar.gz>
- Download MD5 sum: 460824075b51381a4b5f478c60a18165
- Download size: 228 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: less than 0.1 SBU

Asciidoc Dependencies

Optional (runtime)

[docbook-xsl-nons-1.79.2](#), [fop-2.9](#), [libxslt-1.1.42](#), [Lynx-2.9.2](#), [dblatex](#), and [W3m](#)

Installation of Asciidoc

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user asciidoc
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: a2x and asciidoc

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/asciidoc and /usr/lib/python3.12/site-packages/asciidoc-10.2.1.dist-info

Short Descriptions

`a2x` is a toolchain manager for AsciiDoc (converts AsciiDoc text files to other file formats)

`asciidoc` converts an AsciiDoc text file to HTML or DocBook

CSSSelect-1.2.0

Introduction to CSSSelect Module

CSSSelect provides CSS selectors for Python.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/c/cssselect/cssselect-1.2.0.tar.gz>
- Download MD5 sum: 27fbafacce5447cb867acb240d35002a
- Download size: 1.2 MB
- Estimated disk space required: 512 KB (add 488 KB for tests)
- Estimated build time: less than 0.1 SBU

CSSSelect Dependencies

Required

[setuptools_scm-8.1.0](#)

Optional (for testing)

[lxml-5.3.0](#) and [pytest-8.3.2](#)

Installation of CSSSelect

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user cssselect
```

To test the installation issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/cssselect

CacheControl-0.14.0

Introduction to CacheControl Module

CacheControl is a port of the caching algorithms in `httplib2` for use with `requests` session object. It was written because `httplib2`'s better support for caching is often mitigated by its lack of thread safety. The same is true of `requests` in terms of caching.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/psf/cachecontrol/archive/v0.14.0/cachecontrol-0.14.0.tar.gz>
- Download MD5 sum: 221272755063d3827de22e430ad842e6
- Download size: 44 KB
- Estimated disk space required: 344 KB
- Estimated build time: less than 0.1 SBU

CacheControl Dependencies

Required

[msgpack-1.0.8](#) and [requests-2.32.3](#)

Installation of CacheControl

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user cachecontrol
```

This module does not have a working test suite.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells `pip3` to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents `pip3` from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the `install` command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the `install` command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the `install` command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: `doesitcache`

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/CacheControl-0.14.0.dist-info and /usr/lib/python3.12/site-packages/cachecontrol

Short Descriptions

`doesitcache` is an undocumented command line script.

Cython-3.0.11

Introduction to Cython Module

The Cython package provides a compiler for writing C extensions for the Python language.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/cython/cython/releases/download/3.0.11-1/cython-3.0.11.tar.gz>
- Download MD5 sum: 388b85b7c23f501320d19d991b169f5d
- Download size: 2.6 MB
- Estimated disk space required: 142 MB
- Estimated build time: 2.1 SBU

Installation of Cython

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user Cython
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: cygdb, cython, and cythonize

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/Cython, /usr/lib/python3.12/site-packages/Cython-3.0.11.dist-info, and /usr/lib/python3.12/site-packages/pyximport

Short Descriptions

cygdb	is the Cython debugger
cython	is a compiler for code written in the Cython language. It outputs a C/C++ program which can be compiled with a C/C++ compiler
cythonize	is a compiler for code written in the Cython language. It outputs an extension module which is directly importable from Python

dbusmock-0.32.1

Introduction to dbusmock

dbusmock is a Python library useful for writing tests for software which talks to D-Bus services.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/python-dbusmock/python-dbusmock-0.32.1.tar.gz>
- Download MD5 sum: 53b043d9b63247fe49d74c3c299fb1c1
- Download size: 104 KB
- Estimated disk space required: 1.7 MB (with tests)
- Estimated build time: less than 0.1 SBU (add 0.3 SBU for tests)

dbusmock Dependencies

Required

[D-Bus Python-1.3.2](#)

Optional

[pytest-8.3.2](#) (required to run the test suite), [PyGObject-3.48.2](#) (required to run the test suite), [BlueZ-5.77](#) (optional for the test suite), and [UPower-1.90.4](#) (optional for the test suite; if not installed one test will fail)

Installation of dbusmock

Build the package with:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

To install the package run the following as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user python-dbusmock
```

To test the results, issue: `LC_ALL=C pytest`

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory dist.

--no-cache-dir: disables the cache to prevent a warning when installing as the root user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/dbusmock and /usr/lib/python3.12/site-packages/python_dbusmock-0.32.1.dist-info

D-Bus Python-1.3.2

Introduction to D-Bus Python Module

D-Bus Python provides Python bindings to the D-Bus API interface.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://dbus.freedesktop.org/releases/dbus-python/dbus-python-1.3.2.tar.gz>
- Download MD5 sum: 33be8a4a766e1c7c9a377b8f934ce21a
- Download size: 592 KB
- Estimated disk space required: 4.0 MB
- Estimated build time: less than 0.1 SBU

D-Bus Python Dependencies

Required

[dbus-1.14.10](#), [GLib-2.80.4](#), [meson_python-0.16.0](#), and [patchelf-0.18.0](#)

Installation of D-Bus Python

Build the D-Bus Python module by running the following command:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now install the module as the root user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user dbus-python
```

Command Explanations

-w dist: builds the appropriate "wheel" for this module in the directory dist.

--no-build-isolation: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

--no-deps: prevents pip3 from building wheels for the project's dependencies.

--no-index: ignores the package index (only looking at --find-links URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory dist.

--no-cache-dir: disables the cache to prevent a warning when installing as the root user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/dbus and /usr/lib/python3.12/site-packages/dbus_python-1.3.2.egg-info

docutils-0.21.2

Introduction to docutils

docutils is a set of Python modules and programs for processing plaintext docs into formats such as HTML, XML, or LaTeX.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/d/docutils/docutils-0.21.2.tar.gz>
- Download MD5 sum: c4064e1e0e3cd142951fd2b95b830874
- Download size: 2.1 MB
- Estimated disk space required: 10 MB
- Estimated build time: less than 0.1 SBU

Installation of docutils

First, remove some files installed in a previous version that are no longer valid. As the root user:

```
for f in /usr/bin/rst*.py; do
    rm -fv /usr/bin/${basename $f}.py
done
```

To build the Python 3 applications, run the following command:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

To install the Python applications run the following as the root user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user docutils
```

To test the installation, issue: `test/alltests.py`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory dist.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at --find-links URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory dist.

--no-cache-dir: disables the cache to prevent a warning when installing as the root user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: docutils, rst2html4, rst2html5, rst2html, rst2latex, rst2man, rst2odt_prepstyles, rst2odt, rst2pseudoxml, rst2s5, rst2xetex, rst2xml, and rstpep2html

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/docutils{-0.21.2.dist-info}

Short Descriptions

docutils	converts documents into various formats
rst2html	generates (X)HTML documents from standalone reStructuredText sources
rst2html4	generates (X)HTML documents from standalone reStructuredText sources
rst2html5	generates HTML5 documents from standalone reStructuredText sources
rst2latex	generates LaTeX documents from standalone reStructuredText sources
rst2man	generates plain unix manual documents from standalone reStructuredText sources
rst2odt	generates OpenDocument/OpenOffice/ODF documents from standalone reStructuredText sources
rst2odt_prepstyles	Fix a word-processor-generated styles.odt for odtwriter use
rst2pseudoxml	generates pseudo-XML from standalone reStructuredText sources (for testing purposes)
rst2s5	generates S5 (X)HTML slideshow from standalone reStructuredText sources
rst2xetex	generates LaTeX documents from standalone reStructuredText sources for compilation with the Unicode-aware TeX variants XeLaTeX or LuaLaTeX
rst2xml	generates Docutils-native XML from standalone reStructuredText sources
rstpep2html	generates (X)HTML from reStructuredText-format PEP files

Doxypypy-0.8.8.7

Introduction to Doxypypy Module

The Doxypypy package is a doxygen filter for python.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/d/doxypypy/doxypypy-0.8.8.7.tar.gz>
- Download MD5 sum: 5773d0a7882df900cbda8ee5107e1ced
- Download size: 45 KB
- Estimated disk space required: 2.1 MB
- Estimated build time: less than 0.1 SBU

Doxypypy Dependencies

Required

[chardet-5.2.0](#) and [Doxygen-1.12.0](#) (at run time)

Installation of Doxypypy

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user doxypypy
```

This package does not come with a working test suite.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: doxypypy

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/doxypypy and /usr/lib/python3.12/site-packages/doxypypy-0.8.8.7.dist-info

Short Descriptions

`doxypypy` filters Python code for use with Doxygen, using a syntax-aware approach.

Doxyqml-0.5.3

Introduction to Doxyqml Module

The Doxyqml package allows using Doxygen to document QML classes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/d/doxyqml/doxyqml-0.5.3.tar.gz>
- Download MD5 sum: 3d394a0d896721e27beb62bf032f7f43
- Download size: 28 KB
- Estimated disk space required: 652 KB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Doxyqml Dependencies

Required (Run Time)

[Doxygen-1.12.0](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of Doxyqml

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user doxyqml
```

To test the installation, run:

```
pytest
```

The `test_qml/class.py` test is known to fail.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: doxyqml

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/doxyqml and /usr/lib/python3.12/site-packages/doxyqml-0.5.3.dist-info

Short Descriptions

`doxyqml` is a Doxygen input filter for QML files.

Gi-DocGen-2024.1

Introduction to Gi-DocGen Module

Gi-DocGen is a document generator for GObject-based libraries. GObject is the base type system of the GNOME project. Gi-DocGen reuses the introspection data generated by GObject-based libraries to generate the API reference of these libraries, as well as other ancillary documentation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/g/gi-docgen/gi_docgen-2024.1.tar.gz
- Download MD5 sum: 9fb8e8d8536bc9b47060415370a17fc7
- Download size: 2.4 MB
- Estimated disk space required: 17 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Gi-DocGen Dependencies

Required

[Markdown-3.6](#), [packaging-24.1](#), [Pygments-2.18.0](#), and [typogrify-2.0.7](#)

Optional

[Graphviz-12.1.0](#) (runtime, for visualizing class hierarchy graphs) and [pytest-8.3.2](#) (for testing)

Installation of gi-docgen

Silence a warning from the Graphviz `dot` utility which will cause documentation of various packages fail to build if Graphviz is installed but built without Pango, and `gi-docgen` is invoked with the `--fatal-warnings` option:

```
sed -i '/if err:/s/err/proc.returncode/' gidocgen/utils.py
```

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user gi-docgen
```

To test the installation, issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: gi-docgen

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/gidocgen and /usr/lib/python3.12/site-packages/gi_docgen-2024.1.dist-info

Short Descriptions

gi-docgen manages documentation for libgobject based libraries.

html5lib-1.1

Introduction to html5lib Module

html5lib provides a pure-python library for parsing HTML. It is designed to conform to the WHATWG HTML specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/h/html5lib/html5lib-1.1.tar.gz>
- Download MD5 sum: 6748742e2ec4cb99287a6bc82bcfe2b0
- Download size: 268 KB
- Estimated disk space required: 3.9 MB
- Estimated build time: less than 0.1 SBU

html5lib Dependencies

Required

[six-1.16.0](#) and [webencodings-0.5.1](#)

Installation of html5lib

To build the Python 3 module, run:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

To install the module, issue the following command as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user html5lib
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/html5lib and /usr/lib/python3.12/site-packages/html5lib-1.1.dist-info

Ixml-5.3.0

Introduction to Ixml Module

Ixml provides Python bindings for [libxslt-1.1.42](#) and [libxml2-2.13.3](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/l/xml/xml-5.3.0.tar.gz>
- Download MD5 sum: d3e6f800f4b73f2756663cc4a76df5e0
- Download size: 3.6 MB
- Estimated disk space required: 90 MB
- Estimated build time: 0.9 SBU

Ixml Dependencies

Required

[libxslt-1.1.42](#)

Installation of Ixml

To build the Python 3 module, run:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

To install the module, issue the following command as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user lxml
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/lxml and /usr/lib/python3.12/site-packages/lxml-5.3.0.dist-info

Mako-1.3.5

Introduction to Mako Module

Mako is a Python module that implements hyperfast and lightweight templating for the Python platform.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/M/Mako/Mako-1.3.5.tar.gz>
- Download MD5 sum: 0cf9ef37c1f8ffd453ef2b2a3a5573b3
- Download size: 384 KB
- Estimated disk space required: 2.9 MB (add 1.1 MB for tests)
- Estimated build time: less than 0.1 SBU (with tests)

Mako Dependencies

Optional (for Testing)

[pytest-8.3.2](#)

Installation of Mako

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Install the module as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user Mako
```

To test the installation, issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: mako-render

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/mako and /usr/lib/python3.12/site-packages/Mako-1.3.5.dist-info

Short Descriptions

mako-render renders a template

NumPy-2.1.0

Introduction to NumPy Module

NumPy is the fundamental package for scientific computing with Python.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): [https://files.pythonhosted.org/packages/source/n/numpy\(numpy-2.1.0.tar.gz](https://files.pythonhosted.org/packages/source/n/numpy(numpy-2.1.0.tar.gz)
- Download MD5 sum: 4cb2230ffa1cc41329ae29bd69ee08de
- Download size: 18 MB
- Estimated disk space required: 104 MB (add 48 MB for tests)
- Estimated build time: 0.6 SBU (add 1.1 SBU for tests)

NumPy Dependencies

Required

[cython-3.0.11](#), [meson_python-0.16.0](#), and [pyproject-metadata-0.8.0](#)

Optional

fortran from [GCC-14.2.0](#), [lapack and blas](#), and [openblas](#)

Optional (for testing)

[attrs-24.2.0](#), [pytest-8.3.2](#), [pytz-2024.1](#), and [hypothesis](#)

Installation of NumPy

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir -Csetup-args=-Dallow-noblas=true $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user numpy
```

The installation can be tested with the following commands:

```
mkdir -p test
cd test
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate &&
pip3 install hypothesis
```

```
python -c "import numpy, sys; sys.exit(numpy.test() is False)"  
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: f2py, f2py3 and f2py3.12 (3 copies of the same script)

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/numpy

Short Descriptions

`f2py` is the Fortran to Python interface generator utility.

Packaging-24.1

Introduction to Packaging Module

The Packaging library provides utilities that implement the interoperability specifications which have clearly one correct behaviour (PEP440) or benefit greatly from having a single shared implementation (PEP425). This includes utilities for version handling, specifiers, markers, tags, and requirements.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/packaging/packaging-24.1.tar.gz>
- Download MD5 sum: 99b97d1f30017a62a2aae777a14782d0
- Download size: 148 KB
- Estimated disk space required: 2.6 MB (add 20 MB for tests)
- Estimated build time: less than 0.1 SBU (0.2 SBU for tests)

Packaging Dependencies

Optional (for testing)

`pytest-8.3.2` and `pretend`

Installation of Packaging

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user packaging
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependency is not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install pretend
python3 /usr/bin/pytest
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/packaging and /usr/lib/python3.12/site-packages/packaging-24.1.dist-info

ply-3.11

Introduction to ply Module

The ply package is a 100% Python implementation of the common parsing tools 'lex' and 'yacc'.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/ply/ply-3.11.tar.gz>
- Download MD5 sum: 6465f602e656455affcd7c5734c638f8
- Download size: 156 KB
- Estimated disk space required: 2.2 MB
- Estimated build time: less than 0.1 SBU

Installation of ply

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user ply
```

This package does not come with a working test suite.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/ply and /usr/lib/python3.12/site-packages/ply-3.11.dist-info

psutil-6.0.0

Introduction to psutil Module

psutil (process and system utilities) is a cross-platform library for retrieving information on running processes and system utilization (CPU, memory, disks, network, sensors) in Python. It is useful mainly for system monitoring, profiling and limiting process resources and management of running processes. It implements many functionalities offered by classic UNIX command line tools such as ps, top, iotop, lsof, netstat, ifconfig, free and others.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/psutil/psutil-6.0.0.tar.gz>
- Download MD5 sum: 5874bd773d2fe7da3c0817424f383033
- Download size: 500 KB
- Estimated disk space required: 4.3 MB
- Estimated build time: less than 0.1 SBU

Installation of psutil

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user psutil
```

To test the results, issue:

```
mkdir empty &&
(cd empty; python3 -m psutil.tests)
```

Two tests named `test_disk_usage` and `test_io_counters` are known to fail.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/psutil-6.0.0.dist-info

Py3c-1.4

Introduction to Py3c

Py3c helps you port C extensions to Python 3. It provides a detailed guide, and a set of macros to make porting easy and reduce boilerplate.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/encukou/py3c/archive/v1.4/py3c-1.4.tar.gz>
- Download MD5 sum: 53029afde7e0cf8672a2d69d378a0fcf
- Download size: 47 KB
- Estimated disk space required: 608 KB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Installation of Py3c

The Py3c package is a headers-only package and because of that, no configuration and compilation is required.

To test the package, issue:

```
make test-python3 &&
```

```
make test-python3-cpp
```

To install the Python 3 module, run the following command as the `root` user:

```
make prefix=/usr install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/include/py3c

PyAtSpi2-2.46.1

Introduction to PyAtSpi2

The PyAtSpi2 package contains Python bindings for the core components of the GNOME Accessibility.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/pyatspi/2.46/pyatspi-2.46.1.tar.xz>
- Download MD5 sum: 0f2bfafe055457cb0c2389fd46d1ad70
- Download size: 316 KB
- Estimated disk space required: 3.9 MB
- Estimated build time: less than 0.1 SBU

PyAtSpi2 Dependencies

Required

[PyGObject-3.48.2](#)

Recommended

[at-spi2-core-2.52.0](#)

Installation of PyAtSpi2

To build PyAtSpi2 as a Python 3 module, run the following commands:

```
./configure --prefix=/usr --with-python=/usr/bin/python3
```

This package does not come with a test suite.

To install the Python 3 module, run the following command as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pyatspi

PyCairo-1.26.1

Introduction to PyCairo Module

PyCairo provides Python bindings to Cairo.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/pygobject/pycairo/releases/download/v1.26.1/pycairo-1.26.1.tar.gz>
- Download MD5 sum: 36504ac01533ae14f0d2337516bbae2e
- Download size: 340 KB
- Estimated disk space required: 3.1 MB
- Estimated build time: less than 0.1 SBU

PyCairo Dependencies

Required

[Cairo-1.18.0](#)

Optional

[Hypothesis](#) and [pytest-8.3.2](#) (for tests)

Installation of PyCairo

Install PyCairo for Python3 by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To run the tests, this package requires the optional pytest module. If it is installed, run the tests by running `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Program: None

Installed Library: None

Installed Directories: /usr/include/pycairo, /usr/lib/python3.12/site-packages/cairo, and /usr/lib/python3.12/site-packages/pycairo-1.26.1.egg-info

pygdbmi-0.11.0.0

Introduction to pygdbmi Module

GDB/MI is a line based machine oriented text interface to GDB and is activated by specifying using the `--interpreter` command line option (see Mode Options). It is specifically intended to support the development of systems which use the debugger as just one small component of a larger system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pygdbmi/pygdbmi-0.11.0.0.tar.gz>
- Download MD5 sum: 34b1812e77469c6206002b3929798cab
- Download size: 28 KB
- Estimated disk space required: 280 MB
- Estimated build time: less than 0.1 SBU

Installation of pygdbmi

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pygdbmi
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pygdbmi-0.11.0.0.dist-info

Pygments-2.18.0

Introduction to Pygments Module

Pygments is a general syntax highlighter written in Python, for more than 300 languages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/P/Pygments/pygments-2.18.0.tar.gz>
- Download MD5 sum: fbb7976df756f832f3391008778e8999
- Download size: 4.7 MB
- Estimated disk space required: 52 MB
- Estimated build time: less than 0.1 SBU

Pygments Dependencies

Required

[hatchling-1.25.0](#)

Optional (for testing)

[pytest-8.3.2](#) and [wcag-contrast-ratio](#)

Installation of Pygments

Build the Python 3 module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

To install the package run the following as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user Pygments
```

To test the installation, make sure [pytest-8.3.2](#) is installed and run:

```
python3 -m venv --system-site-packages testenv &&  
testenv/bin/pip3 install wcag-contrast-ratio &&  
testenv/bin/python -m pytest
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: pygmentize

Installed Library: None

Installed Directories: /usr/lib/python3.12/site-packages/pygments and /usr/lib/python3.12/site-packages/Pygments-2.18.0.dist-info

Short Descriptions

`pygmentize` highlights an input file and writes the result to an output file

PyGObject-3.48.2

Introduction to PyGObject3 Module

PyGObject3 provides Python bindings to the GObject class from GLib.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/pygobject/3.48/pygobject-3.48.2.tar.xz>
- Download MD5 sum: fe1cb825adc8a49d1629b97b7d26dffc
- Download size: 544 KB

- Estimated disk space required: 7.0 MB (add 2.5 MB for tests)
- Estimated build time: 0.2 SBU (with tests)

PyGObject3 Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection)

Recommended

[PyCairo-1.26.1](#)

Optional (for the tests)

[GTK-4.14.5](#), [pep8](#), [pyflakes](#), and [pytest-8.3.2](#)

Installation of PyGObject3

First, remove two faulty tests:

```
mv -v tests/test_gdbus.py{,.nouse}      &&
mv -v tests/test_overrides_gtk.py{,.nouse}
```

Install pygobject3 by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`. An already active graphical session with a bus address is necessary to run the tests. Another round of tests may report ERROR if [GTK-4.14.5](#) is not installed.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D pycairo=disabled`: use this switch if you do not have [PyCairo-1.26.1](#) installed.

Contents

Installed Programs: None

Installed Library: /usr/lib/python3.12/site-packages/gi/_gi{,_cairo}.cpython-312-<arch>-linux-gnu.so

Installed Directories: /usr/include/pygobject-3.0 and /usr/lib/python3.12/site-packages/{gi,pygtkcompat}

pyparsing-3.1.2

Introduction to pyparsing Module

The pyparsing module is an alternative approach to creating and executing simple grammars, vs. the traditional lex/yacc approach, or the use of regular expressions. It provides a library of classes that client code uses to construct the grammar directly in Python code.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pyparsing/pyparsing-3.1.2.tar.gz>
- Download MD5 sum: 2bfafdb2d02d19ca4a3dfd02a9dbdfa7
- Download size: 872 KB
- Estimated disk space required: 3.3 MB (add 27 MB for tests)
- Estimated build time: less than 0.1 SBU (0.3 SBU for tests)

pyparsing Dependencies

Optional

[railroad-diagrams](#) (also needed for tests)

Optional (for testing)

[pytest-8.3.2](#)

Installation of pyparsing

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pyparsing
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependency is not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate &&
pip3 install railroad-diagrams &&
python3 /usr/bin/pytest
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pyparsing and /usr/lib/python3.12/site-packages/pyparsing-3.1.2.dist-info

pySerial-3.5

Introduction to pySerial Module

The pySerial module encapsulates access to the serial port.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pyserial/pyserial-3.5.tar.gz>
- Download MD5 sum: 1cf25a76da59b530dbfc2cf99392dc83
- Download size: 156 KB
- Estimated disk space required: 2.1 MB (add 0.2 MB for tests)
- Estimated build time: less than 0.1 SBU

pySerial Dependencies

Required

[setuptools_scm-8.1.0](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of pySerial

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pyserial
```

To test the installation issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: pyserial-miniterm and pyserial-ports

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pyserial-3.5.dist-info and /usr/lib/python3.12/site-packages/serial

Short Descriptions

`pyserial-miniterm` is a console application that provides a small terminal application. It may inherit terminal features from the terminal in which it is run.
`pyserial-ports` lists available ports.

Pytest-8.3.2

Introduction to Pytest Module

The Pytest framework makes it easy to write small, readable tests, and can scale to support complex functional testing for applications and libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pytest/pytest-8.3.2.tar.gz>
- Download MD5 sum: 9e6443870618f16cad55527b8302d134
- Download size: 1.4 MB
- Estimated disk space required: 43 MB (with tests)
- Estimated build time: 1.3 SBU (with tests)

Pytest Dependencies

Required

[iniconfig-2.0.0](#), [packaging-24.1](#), and [pluggy-1.5.0](#)

Recommended

[setuptools_scm-8.1.0](#)

Optional (for testing)

[attrs-24.2.0](#), [Pygments-2.18.0](#), [requests-2.32.3](#), [argcomplete](#), [hypothesis](#), [mock](#), and [xmlschema](#)

Installation of Pytest

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pytest
```

The installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate &&
pip3 install pytest[dev] xmlschema hypothesis &&
```

```
python3 /usr/bin/pytest
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the `install` command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the `install` command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: pytest and py.test (different files but with same content)

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/_pytest, /usr/lib/python3.12/site-packages/pytest, and /usr/lib/python3.12/site-packages/pytest-8.3.2.dist-info

Short Descriptions

`pytest` sets up, manages, and/or runs tests in python module source directories

PyXDG-0.28

Introduction to PyXDG Module

PyXDG is a Python library to access freedesktop.org standards.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP):
<https://files.pythonhosted.org/packages/b0/25/7998cd2dec731acbd438fbf91bc619603fc5188de0a9a17699a7810.28.tar.gz>
- Download MD5 sum: d9a1d04fe60c956f5e3b9de3b4ef4722
- Download size: 76 KB
- Estimated disk space required: 808 KB
- Estimated build time: less than 0.1 SBU

Installation of PyXDG

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Install the module as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pyxdg
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Directory: /usr/lib/python3.12/site-packages/xdg and /usr/lib/python3.12/site-packages/pyxdg-0.28.dist-info

PyYAML-6.0.2

Introduction to PyYAML Module

PyYAML is a Python module that implements the next generation YAML parser and emitter.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/P/PyYAML/pyyaml-6.0.2.tar.gz>
- Download MD5 sum: 9600ee49b2b4e1a0237cf4173b6dc594
- Download size: 128 KB
- Estimated disk space required: 8.5 MB
- Estimated build time: 0.1 SBU

PyYAML Dependencies

Required

[cython-3.0.11](#) and [libyaml-0.2.5](#)

Optional

[pytest-8.3.2](#) (for testing)

Installation of PyYAML

Build PyYAML with the following command:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, install the module as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user PyYAML
```

To test the results, issue: `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/PyYAML-0.2.5.dist-info

Recommonmark-0.7.1

Introduction to Recommonmark Module

Recommonmark is a docutils-compatibility bridge to CommonMark. It allows writing CommonMark inside of Docutils and Sphinx projects. Recommonmark is now deprecated in favor of [MyST-Parser](#). Unfortunately, [LLVM-18.1.7](#) still depends on this module.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/r/recommonmark/recommonmark-0.7.1.tar.gz>
- Download MD5 sum: 3c550a76eb62006bf007843a9f1805bb
- Download size: 34 KB
- Estimated disk space required: 680 KB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Recommonmark Dependencies

Required

[commonmark-0.9.1](#) and [sphinx-8.0.2](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of Recommonmark

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user recommonmark
```

Recommonmark is now deprecated in favor of [MyST-Parser](#). For this reason, a lot of tests have to be disabled because they do not pass with recent versions of Sphinx. The installation can be partially tested with the following command:

```
pytest -k 'not (test_integration or test_code or test_headings or test_image or test_links or test_lists)'
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: cm2html, cm2latex, cm2man, cm2pseudoxml, cm2xetex, and cm2xml

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/recommonmark and /usr/lib/python3.12/site-packages/recommonmark-0.7.1.dist-info

Short Descriptions

<code>cm2html</code>	generates a html document from markdown sources
<code>cm2latex</code>	generates a latex document from markdown sources
<code>cm2man</code>	generates a manpage from markdown sources
<code>cm2pseudoxml</code>	generates a pseudo-XML document from markdown sources
<code>cm2xetex</code>	generates a xetex document from markdown sources
<code>cm2xml</code>	generates an XML document from markdown sources

Requests-2.32.3

Introduction to Requests Module

The Requests package is an elegant and simple HTTP library for Python, built for human beings. It allows sending HTTP/1.1 requests extremely easily.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/r/requests/requests-2.32.3.tar.gz>
- Download MD5 sum: fa3ee5ac3f1b3f4368bd74ab530d3f0f
- Download size: 132 KB
- Estimated disk space required: 1.1 MB (add 139 MB for tests)
- Estimated build time: less than 0.1 SBU (0.9 SBU for tests)

Additional Download

- Recommended patch: https://www.linuxfromscratch.org/patches/blfs/12.2/requests-2.32.3-use_system_certs-1.patch

Requests Dependencies

Required

[charset-normalizer-3.3.2](#), [idna-3.7](#), and [urllib3-2.2.2](#)

Recommended

[make-ca-1.14](#), with [p11-kit-0.25.5](#) (both needed to use system https: certificates, removing an unnecessary dependency on the Certifi module).

Optional

[PySocks](#) (also needed for testing)

Optional (for testing)

[pytest-8.3.2](#), [Flask<2](#), [httpbin](#), [MarkupSafe<2.1](#), [pytest-mock](#), [pytest-httpbin](#), [sphinx<5](#), [trustme](#), and [Werkzeug<2](#)

Installation of Requests

First apply a patch so that the same environment variable already used to point to system certificates in Python3 following the installation of make-ca can also be used by this module:

```
patch -Np1 -i .../requests-2.32.3-use_system_certs-1.patch
```

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user requests
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependencies are not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install --force-reinstall sphinx<5 &&
pip3 install pytest-mock \
    werkzeug\<2 \
    flask\<2 \
    pytest-httpbin \
    pysocks \
    trustme &&
pip3 install --force-reinstall MarkupSafe\<2.1 &&
python3 /usr/bin/pytest tests
deactivate
```

Note

See [make-ca-1.14](#) for how to set the environment variable, and for how local copies of the Certifi and Requests modules installed in a virtual environment *will* override the system certificates.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/requests and /usr/lib/python3.12/site-packages/requests-2.32.3.dist-info

Scour-0.38.2

Introduction to Scour Module

Scour is an SVG (Scalable Vector Graphics) optimizer/cleaner that reduces their size by optimizing structure and removing unnecessary data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/scour-project/scour/archive/v0.38.2/scour-0.38.2.tar.gz>
- Download MD5 sum: ae30f52602802f8c7df3a32e1f72b325
- Download size: 100 KB
- Estimated disk space required: 1.3 MB
- Estimated build time: less than 0.1 SBU

Scour Dependencies

Required

[six-1.16.0](#)

Installation of scour

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

To install the module, run the following command as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user scour
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: scour

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/scour and /usr/lib/python3.12/site-packages/scour-0.38.2.dist-info

Short Descriptions

`scour` is a program to optimize and clean SVG files

sentry-sdk-2.13.0

Introduction to `sentry-sdk` Module

The `sentry-sdk` module is the official Python SDK for Sentry.io.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/getsentry/sentry-python/releases/download/2.13.0/sentry_sdk-2.13.0.tar.gz
- Download MD5 sum: d56d6bf0acd0bd0663b420199c9afc29
- Download size: 276 KB
- Estimated disk space required: 3.4 MB
- Estimated build time: less than 0.1 SBU

`sentry-sdk` Dependencies

Required

`certifi-2024.7.4` and `urllib3-2.2.2`

Installation of `sentry-sdk`

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sentry-sdk
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sentry_sdk-2.13.0.dist-info and /usr/lib/python3.12/site-packages/sentry_sdk

six-1.16.0

Introduction to Six Module

Six is a Python 2 to 3 compatibility library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/s/six/six-1.16.0.tar.gz>
- Download MD5 sum: a7c927740e4964dd29b72cebf1429bb
- Download size: 36 KB
- Estimated disk space required: 376 KB
- Estimated build time: less than 0.1 SBU

Installation of Six

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user six
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/six-1.16.0.dist-info

Sphinx-8.0.2

Introduction to Sphinx Module

The Sphinx package is a set of tools for translating some structured text formats into pretty documentation in various formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/s/sphinx/sphinx-8.0.2.tar.gz>
- Download MD5 sum: 9f07671060e83d92a7a4e33893bbd408
- Download size: 7.8 MB
- Estimated disk space required: 57 MB (with tests)
- Estimated build time: less than 0.1 SBU (add 1.0 SBU for tests)

Sphinx Dependencies

Required

[alabaster-1.0.0](#), [babel-2.16.0](#), [docutils-0.21.2](#), [imagesize-1.4.1](#), [packaging-24.1](#), [Pygments-2.18.0](#), [requests-2.32.3](#), [snowballstemmer-2.2.0](#), [sphinxcontrib-applehelp-2.0.0](#), [sphinxcontrib-devhelp-2.0.0](#), [sphinxcontrib-htmlhelp-2.0.1](#), [sphinxcontrib-jsmath-1.0.1](#), [sphinxcontrib-qthelp-2.0.0](#), and [sphinxcontrib-serializinghtml-2.0.0](#)

Optional (for tests)

[cython-3.0.11](#), [html5lib-1.1](#), [pytest-8.3.2](#), [texlive-20240312](#), [defusedxml](#), and [typing_extensions](#)

Installation of Sphinx

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinx
```

Assuming [cython-3.0.11](#) and [pytest-8.3.2](#) are installed, but the other optional dependencies are not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install sphinx[test]
python3 -m pytest
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: sphinx-apidoc, sphinx-autogen, sphinx-build, and sphinx-quickstart

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinx and /usr/lib/python3.12/site-packages/sphinx-8.0.2.dist-info

Short Descriptions

<code>sphinx-apidoc</code>	creates an reST file from python modules and packages
<code>sphinx-autogen</code>	generates ReStructuredText from special directives contained in given input files
<code>sphinx-build</code>	generates documentation in various formats from ReStructuredText source files
<code>sphinx-quickstart</code>	generates required files for a sphinx project

Sphinx_rtd_theme-2.0.0

Introduction to Sphinx_rtd_theme Module

The `sphinx_rtd_theme` module is a Sphinx theme designed to provide a great reader experience for documentation users on both desktop and mobile devices. This theme is used primarily on Read the Docs but can work with any Sphinx project.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/s/sphinx_rtd_theme/sphinx_rtd_theme-2.0.0.tar.gz
- Download MD5 sum: 8ad74a8e9ca706ed77117be0e9eed7e8
- Download size: 2.7 MB
- Estimated disk space required: 36 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Sphinx_rtd_theme Dependencies

Required

[sphinx-8.0.2](#) and [sphinxcontrib-jquery-4.1](#)

Optional (for tests)

[pytest-8.3.2](#) and [readthedocs-sphinx-ext](#)

Installation of Sphinx_rtd_theme

First, tell the package that newer [docutils-0.21.2](#) and [sphinx-8.0.2](#) versions are safe to use:

```
sed -e 's/0.21/0.22/' \
    -e 's/8$/9/' \
    -i setup.cfg
```

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinx_rtd_theme
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependency is not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate &&
pip3 install readthedocs-sphinx-ext &&
python3 /usr/bin/pytest
deactivate
```

Several tests return warnings because they use a deprecated sphinx API.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinx_rtd_theme and /usr/lib/python3.12/site-packages/sphinx_rtd_theme-2.0.0.dist-info

Python Dependencies

Introduction to Python Dependencies

Python modules listed in [Python Modules](#) have dependencies that are not referenced by other packages in BLFS. These dependencies are listed here. They will not get updated on regular basis, unless a more recent version is needed.

Important

In BLFS, we normally build and install Python 3 modules with `pip3`. Please take care that the `pip3 install` commands in the book should be run as `root` unless it's for a Python virtual environment. Running `pip3 install` as a non-`root` user may seem to work fine, but it will cause the installed module to be inaccessible by other users.

`pip3 install` will not reinstall an already installed module by default. For using the `pip3 install` command to upgrade a module (for example, from meson-0.61.3 to meson-0.62.0), insert `--upgrade` into the command line. If it's really necessary to downgrade a module or reinstall the same version for some reason, insert `--force-reinstall` into the command line.

- [alabaster-1.0.0](#)
- [attrs-24.2.0](#)
- [babel-2.16.0](#)
- [certifi-2024.7.4](#)
- [chardet-5.2.0](#)
- [charset-normalizer-3.3.2](#)
- [commonmark-0.9.1](#)
- [editable-0.5](#)
- [hatchling-1.25.0](#)
- [hatch-fancy-pypi-readme-24.1.0](#)
- [hatch_vcs-0.4.0](#)
- [idna-3.7](#)
- [imagesize-1.4.1](#)
- [iniconfig-2.0.0](#)
- [Markdown-3.6](#)
- [meson_python-0.16.0](#)
- [msgpack-1.0.8](#)
- [pathspec-0.12.1](#)
- [pluggy-1.5.0](#)
- [pyproject-metadata-0.8.0](#)
- [pytz-2024.1](#)
- [setuptools_scm-8.1.0](#)
- [smartypants-2.0.1](#)
- [snowballstemmer-2.2.0](#)

- [sphinxcontrib-applehelp-2.0.0](#)
- [sphinxcontrib-devhelp-2.0.0](#)
- [sphinxcontrib-htmlhelp-2.0.1](#)
- [sphinxcontrib-jquery-4.1](#)
- [sphinxcontrib-jsmath-1.0.1](#)
- [sphinxcontrib-qthelp-2.0.0](#)
- [sphinxcontrib-serializinghtml-2.0.0](#)
- [trove-classifiers-2024.7.2](#)
- [typogrify-2.0.7](#)
- [urllib3-2.2.2](#)
- [webencodings-0.5.1](#)

Alabaster-1.0.0

Introduction to Alabaster Module

The Alabaster package is a theme for the sphinx documentation system. Although developed separately, it is the default theme for sphinx.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/a/alabaster/alabaster-1.0.0.tar.gz>
- Download MD5 sum: c6c2173e5565fb12f08bef410ea50f72
- Download size: 24 KB
- Estimated disk space required: 160 KB
- Estimated build time: less than 0.1 SBU

Installation of Alabaster

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user alabaster
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/alabaster and /usr/lib/python3.12/site-packages/alabaster-1.0.0.dist-info

Attrs-24.2.0

Introduction to Attrs Module

The Attrs package is a python module that allows for writing attributes without extra boilerplate.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/a/attrs/attrs-24.2.0.tar.gz>
- Download MD5 sum: 031a41b703d7fc47ec204d0369b68dc
- Download size: 776 KB
- Estimated disk space required: 7 MB (add 78 MB for tests)
- Estimated build time: less than 0.1 SBU (0.2 SBU for tests)

Attrs Dependencies

Required

[hatch-fancy-pypi-readme-24.1.0](#) and [hatch_vcs-0.4.0](#)

Optional (for testing)

[pytest-8.3.2](#), [cloudpickle](#), [hypothesis](#), [Pympler](#), [mypy](#), [pytest-mypy-plugins](#), [pytest-xdist\[psutil\]](#), and [zope.interface](#)

Installation of Attrs

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user attrs
```

To test the installation, run:

```
python3 -m venv --system-site-packages testenv      &&
testenv/bin/pip3 install 'attrs[tests]'           &&
PATH=$PWD/testenv/bin:$PATH testenv/bin/python -m pytest
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory dist.

--no-cache-dir: disables the cache to prevent a warning when installing as the root user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/attr, /usr/lib/python3.12/site-packages/attrs, and /usr/lib/python3.12/site-packages/attrs-24.2.0.dist-info

babel-2.16.0

Introduction to Babel Module

The Babel package is an integrated collection of utilities that assist in internationalizing and localizing Python applications, with an emphasis on web-based applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/B/Babel/babel-2.16.0.tar.gz>
- Download MD5 sum: 1dbf17974d7ccc9950757ed3c9618db0
- Download size: 9.0 MB
- Estimated disk space required: 90 MB (add 30 MB for tests)
- Estimated build time: less than 0.1 SBU (0.1 SBU for tests)

Babel Dependencies

Required

[pytz-2024.1](#)

Optional (for testing)

[pytest-8.3.2](#), [Python-3.12.5](#) (with the sqlite module), [freezegun==0.3.12](#), and [pytest-cov](#)

Installation of Babel

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the root user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user Babel
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependencies are not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate &&
pip3 install pytest-cov freezegun==0.3.12 &&
```

```
python3 /usr/bin/pytest
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: `pybabel`

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/babel and /usr/lib/python3.12/site-packages/Babel-2.16.0.dist-info

Short Descriptions

`pybabel` is a command-line interface for working with message catalogs

certifi-2024.7.4

Introduction to Certifi Module

The certifi module provides Mozilla's carefully curated collection of Root Certificates for validating the trustworthiness of SSL certificates while verifying the identity of TLS hosts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/c/certifi/certifi-2024.7.4.tar.gz>
- Download MD5 sum: ecf1d20e4c505fc07c8f421063d04103
- Download size: 164 KB
- Estimated disk space required: 844 KB
- Estimated build time: less than 0.1 SBU

Installation of certifi

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user certifi
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/certifi and /usr/lib/python3.12/site-packages/certifi-2024.7.4.dist-info

Chardet-5.2.0

Introduction to chardet Module

Chardet is a universal character encoding detector.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/c/chardet/chardet-5.2.0.tar.gz>
- Download MD5 sum: cc2d8cc9a751641463b4f7cfecad2ffa
- Download size: 2 MB
- Estimated disk space required: 12 MB (add 1.1 MB for tests)
- Estimated build time: less than 0.1 SBU (0.3 SBU for tests)

Chardet Dependencies

Optional (for testing)

[pytest-8.3.2](#)

Installation of Chardet

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user chardet
```

To test the installation issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: `chardetect`

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/chardet and /usr/lib/python3.12/site-packages/chardet-5.2.0.dist-info

Short Descriptions

`chardetect` is a Universal Character Encoding Detector

Charset-normalizer-3.3.2

Introduction to charset-normalizer Module

The charset-normalizer library helps with reading text from an unknown character encoding.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/c/charset-normalizer/charset-normalizer-3.3.2.tar.gz>
- Download MD5 sum: 0a4019908d9e50ff13138e8a794d9e2b
- Download size: 104 KB
- Estimated disk space required: 1.1 MB (add 19 MB for tests)
- Estimated build time: less than 0.1 SBU (with tests)

Charset-normalizer Dependencies

Optional (for testing)

`pytest-8.3.2`, `Python-3.12.5` (rebuilt after installing `SQLite-3.46.1`), and `pytest-cov`

Installation of Charset-normalizer

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user charset-normalizer
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependencies are not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install pytest-cov
python3 /usr/bin/pytest
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: `normalizer`

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/charset_normalizer and /usr/lib/python3.12/site-packages/charset_normalizer-3.3.2.dist-info

Short Descriptions

`normalizer` is a Universal Charset Detector (discovers originating encoding and normalizes text to unicode)

Commonmark-0.9.1

Introduction to Commonmark Module

Commonmark Python parser for the CommonMark Markdown specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/c/commonmark/commonmark-0.9.1.tar.gz>
- Download MD5 sum: cd1dc70c4714d9ed4117a40490c25e00
- Download size: 94 KB
- Estimated disk space required: 1.3 MB (add 30 MB for tests)
- Estimated build time: less than 0.1 SBU (0.1 SBU for tests)

Commonmark Dependencies

Optional (for testing)

[pytest-8.3.2](#), [flake8](#), and [hypothesis](#)

Installation of Commonmark

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user commonmark
```

Assuming [pytest-8.3.2](#) is installed and the other optional dependency is not, the installation can be tested with:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install hypothesis
python3 /usr/bin/pytest commonmark/tests/unit_tests.py
python3 commonmark/tests/run_spec_tests.py
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: cmark

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/commonmark and /usr/lib/python3.12/site-packages/commonmark-0.9.1.dist-info

Short Descriptions

`cmark` processes Markdown according to the CommonMark specification

Editables-0.5

Introduction to Editables Module

Editables is python library for creating “editable wheels.”

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/e/editables/editables-0.5.tar.gz>
- Download MD5 sum: 520de8c3a9dc5dfb2b365d104541c9de
- Download size: 4.6 KB
- Estimated disk space required: 180 KB
- Estimated build time: less than 0.1 SBU

Editables Dependencies

Optional (for testing)

[pytest-8.3.2](#)

Installation of Editables

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user editables
```

To test the installation, issue: `pytest`.

Command Explanations

`--w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/editables and /usr/lib/python3.12/site-packages/editables-0.5.dist-info

Hatchling-1.25.0

Introduction to Hatchling Module

Hatchling is an extensible, standards compliant build backend for python modules.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/h/h/hatchling/hatchling-1.25.0.tar.gz>
- Download MD5 sum: ac265366ccd17365db319dc95a9e5bc5
- Download size: 60 KB
- Estimated disk space required: 2.3 MB
- Estimated build time: less than 0.1 SBU

Hatchling Dependencies

Required

[editables-0.5](#), [packaging-24.1](#), [pathspec-0.12.1](#), [pluggy-1.5.0](#), and [trove-classifiers-2024.7.2](#)

Installation of Hatchling

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a working test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user hatchling
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: hatchling

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/hatchling and /usr/lib/python3.12/site-packages/hatchling-1.25.0.dist-info

Short Descriptions

`hatchling` is a python module builder

Hatch-Fancy-Pypi-Readme-24.1.0

Introduction to Hatch-Fancy-Pypi-Readme Module

Hatch-Fancy-Pypi-Readme is a hatch plugin filling the readme field into the metadata of a Python module from markup language documentation files of the module.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/h/hatch-fancy-pypi-readme/hatch_fancy_pypi_readme-24.1.0.tar.gz
- Download MD5 sum: f5f9e639f066c91f8e623ec6231beae9
- Download size: 32 KB
- Estimated disk space required: 388 KB
- Estimated build time: less than 0.1 SBU

Hatch-Fancy-Pypi-Readme Dependencies

Required

[hatchling-1.25.0](#)

Optional (for testing)

[pytest-8.3.2](#) and [build](#)

Installation of Hatch-Fancy-Pypi-Readme

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user hatch-fancy-pypi-readme
```

To test the installation, make sure [pytest-8.3.2](#) is installed and run:

```
python3 -m venv --system-site-packages testenv &&
testenv/bin/pip3 install 'hatch-fancy-pypi-readme[tests]' &&
testenv/bin/python -m pytest
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/hatch_fancy_pypi_readme and /usr/lib/python3.12/site-packages/hatch_fancy_pypi_readme-24.1.0.dist-info

Hatch_vcs-0.4.0

Introduction to Hatch-vcs Module

Hatch_vcs is a Hatch plugin for versioning with several [VCS](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/h/hatch-vcs/hatch_vcs-0.4.0.tar.gz
- Download MD5 sum: d801fe7c3e5955307748f2790bbb3488
- Download size: 9.9 KB
- Estimated disk space required: 436 KB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Hatch_vcs Dependencies

Required

[hatchling-1.25.0](#) and [setuptools_scm-8.1.0](#)

Optional (for testing)

[git-2.46.0](#) and [pytest-8.3.2](#)

Installation of Hatch_vcs

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user hatch_vcs
```

To test the installation, issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/hatch_vcs and /usr/lib/python3.12/site-packages/hatch_vcs-0.4.0.dist-info

Idna-3.7

Introduction to Idna Module

The Idna module provides support for the Internationalized Domain Names in Applications (IDNA) protocol as specified in RFC 5891.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/i/idna/idna-3.7.tar.gz>
- Download MD5 sum: 31cc572cb7a6519159c927c998c64c79
- Download size: 188 KB
- Estimated disk space required: 1.6 MB (with test)
- Estimated build time: less than 0.1 SBU (with tests)

Idna Dependencies

Optional (for testing)

[pytest-8.3.2](#)

Installation of Idna

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user idna
```

To test the installation, run `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/idna and /usr/lib/python3.12/site-packages/idna-3.7.dist-info

Imagesize-1.4.1

Introduction to Imagesize Module

The imagesize package analyzes image file headers and returns the image size and DPI. It works with JPEG/JPEG 2000/PNG/GIF/TIFF/SVG/Netpbm/WebP formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/i/imagesize/imagesize-1.4.1.tar.gz>
- Download MD5 sum: 5a40586a25c07e1a8f16f6267252c321
- Download size: 1.2 MB
- Estimated disk space required: 1.8 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Imagesize Dependencies

Optional (for testing)

[pytest-8.3.2](#) and [requests-2.32.3](#)

Installation of Imagesize

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user imagesize
```

The tests for this package are known to be broken. To test the installation anyway, run `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/imagesize and /usr/lib/python3.12/site-packages/imagesize-1.4.1.dist-info

Iniconfig-2.0.0

Introduction to Iniconfig Module

Iniconfig is a small and simple INI-file parser module.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/i/iniconfig/iniconfig-2.0.0.tar.gz>
- Download MD5 sum: 3c030b3f51dcc3aca585de05635600e4
- Download size: 4.5 KB
- Estimated disk space required: 168 KB
- Estimated build time: less than 0.1 SBU

Iniconfig Dependencies

Required

[hatch vcs-0.4.0](#)

Installation of Iniconfig

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user iniconfig
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/iniconfig and /usr/lib/python3.12/site-packages/iniconfig-2.0.0.dist-info

Markdown-3.6

Introduction to Markdown Module

Markdown is a Python parser for John Gruber's Markdown specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/M/Markdown/Markdown-3.6.tar.gz>
- Download MD5 sum: 1ee0e93da8bb2fc2b5830d5b225d3b2b
- Download size: 348 KB
- Estimated disk space required: 4.1 MB (add 27 MB for tests)
- Estimated build time: less than 0.1 SBU (with tests)

Markdown Dependencies

Optional (for testing)

[pytest-8.3.2](#), [PyYAML-6.0.2](#), and [coverage](#)

Installation of Markdown

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user Markdown
```

Assuming [pytest-8.3.2](#) and [PyYAML-6.0.2](#) are installed and the other optional dependency is not, the installation can be tested with:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install coverage
python3 /usr/bin/pytest --ignore=tests/test_syntax/extensions/test_md_in_html.py
deactivate
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the `install` command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the `install` command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: `markdown_py`

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/markdown and /usr/lib/python3.12/site-packages/Markdown-3.6.dist-info

Short Descriptions

`markdown_py` converts markdown files to (x)html

Meson_python-0.16.0

Introduction to Meson_python Module

The `Meson_python` module contains a Python build backend (PEP 517) for Meson projects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/m/meson_python/meson_python-0.16.0.tar.gz
- Download MD5 sum: 0278a447d7aef1cf20964aa369cd5d3b
- Download size: 80 KB
- Estimated disk space required: 1.3 MB
- Estimated build time: less than 0.1 SBU

Meson_python Dependencies

Required

[pyproject-metadata-0.8.0](#)

Recommended (Runtime)

[patchelf-0.18.0](#)

Optional (for testing)

[cython-3.0.11](#), [git-2.46.0](#), [pytest-8.3.2](#), [Python-3.12.5](#) (rebuilt after installing [SQLite-3.46.1](#)), [build](#), and [pytest-mock](#)

Installation of Meson_python

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user meson_python
```

To test the installation, make sure [git-2.46.0](#), [patchelf-0.18.0](#), and [pytest-8.3.2](#) are installed, and [Python-3.12.5](#) has been rebuilt after installing [SQLite-3.46.1](#), then issue (`HOME=` prevents the `.gitconfig` file in the home directory from interfering the tests):

```
python3 -m venv --system-site-packages testenv &&
testenv/bin/pip3 install 'meson_python[test]' &&
HOME= testenv/bin/python -m pytest
```

Two tests named `test_missing_version` and `test_pep621` are known to fail with pyproject-metadata-0.8.0 or later.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/meson_python and /usr/lib/python3.12/site-packages/meson_python-0.16.0.dist-info

Msgpack-1.0.8

Introduction to Msgpack Module

Msgpack is an efficient binary serialization format. It lets you exchange data among multiple languages like JSON. But it's faster and smaller. This package provides CPython bindings for reading and writing Msgpack data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/m/msgpack/msgpack-1.0.8.tar.gz>
- Download MD5 sum: 6f4d91b00537fd5069dc6bfc52ae5652
- Download size: 164 KB
- Estimated disk space required: 1.9 MB (add 0.3 MB for tests)
- Estimated build time: less than 0.1 SBU

Msgpack Dependencies

Required

[cython-3.0.11](#) and [setuptools_scm-8.1.0](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of Msgpack

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user msgpack
```

To test the installation issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/msgpack-1.0.8.dist-info and /usr/lib/python3.12/site-packages/msgpack

Pathspec-0.12.1

Introduction to Pathspec Module

Pathspec is a utility library for pattern matching of file paths.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pathspec/pathspec-0.12.1.tar.gz>
- Download MD5 sum: 2b26ad1981bfa23748e115f00085624c
- Download size: 45 KB

- Estimated disk space required: 912 KB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Pathspec Dependencies

Optional (for testing)

[pytest-8.3.2](#)

Installation of Pathspec

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pathspec
```

To test the installation, issue `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pathspec and /usr/lib/python3.12/site-packages/pathspec-0.12.1.dist-info

Pluggy-1.5.0

Introduction to Pluggy Module

The Pluggy package gives users the ability to extend or modify the behaviour of a host program by installing a plugin for that program. The plugin code will run as part of normal program execution, changing or enhancing certain aspects of it. In essence, pluggy enables function hooking so a user can build "pluggable" systems.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pluggy/pluggy-1.5.0.tar.gz>

- Download MD5 sum: ac0870be78ba0ee227a5c3955efeba59
- Download size: 68 KB
- Estimated disk space required: 588 KB (add 18 MB for tests)
- Estimated build time: less than 0.1 SBU (with tests)

Pluggy Dependencies

Recommended

[setuptools_scm-8.1.0](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of Pluggy

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pluggy
```

To test the installation, issue: `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pluggy and /usr/lib/python3.12/site-packages/pluggy-1.5.0.dist-info

Pyproject-Metadata-0.8.0

Introduction to Pyproject-Metadata Module

The Pyproject-Metadata module contains a data class for PEP 621 metadata with support for "core metadata" (PEP 643) generation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/p/pyproject-metadata/pyproject_metadata-0.8.0.tar.gz
- Download MD5 sum: 048ef3f91a5ba7c89a33894cb2c7725d
- Download size: 8 KB
- Estimated disk space required: 124 KB
- Estimated build time: less than 0.1 SBU

Pyproject-Metadata Dependencies

Required

[packaging-24.1](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of Pyproject-Metadata

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pyproject-metadata
```

To test the installation, issue: `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pyproject_metadata and /usr/lib/python3.12/site-packages/pyproject_metadata-0.8.0.dist-info

Pytz-2024.1

Introduction to Pytz Module

The Pytz library brings the IANA tz database into Python. It allows accurate and cross-platform timezone calculations.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/p/pytz/pytz-2024.1.tar.gz>
- Download MD5 sum: b26bb090d1fe96064019bf7068eeb801
- Download size: 312 KB
- Estimated disk space required: 6.1 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Pytz dependencies

Optional (for testing)

[pytest-8.3.2](#)

Installation of Pytz

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user pytz
```

To test the installation, run `pytest`. A few warnings are issued.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the `install` command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the `install` command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/pytz and /usr/lib/python3.12/site-packages/pytz-2024.1.dist-info

SetupTools_scm-8.1.0

Introduction to SetupTools_scm Module

The SetupTools_scm package is used to extract Python package versions from git or hg metadata instead of declaring them.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/s/setupTools_scm/setupTools_scm-8.1.0.tar.gz
- Download MD5 sum: d8046dce093a94dc382b68b45f6a6257
- Download size: 70 KB
- Estimated disk space required: 1.6 MB (with tests)
- Estimated build time: less than 0.1 SBU (add 0.4 SBU for tests)

SetupTools_scm Dependencies

Required

[packaging-24.1](#)

Optional (for testing)

[git-2.46.0](#), [Mercurial-6.8.1](#), [pytest-8.3.2](#), [Sudo-1.9.15p5](#), and [build](#)

Installation of SetupTools_scm

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user setuptools_scm
```

To test the installation, make sure [pytest-8.3.2](#) is installed and run (`HOME=` prevents the `.gitconfig` file in the home directory from interfering the tests):

```
python3 -m venv --system-site-packages testenv &&
testenv/bin/pip3 install build &&
HOME= testenv/bin/python -m pytest
```

If [git-2.46.0](#) and/or [Mercurial-6.8.1](#) are not installed, the tests depending on the missing one(s) will be skipped. Some tests may invoke [Sudo-1.9.15p5](#) and request a password.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/setuptools_scm and /usr/lib/python3.12/site-packages/setuptools_scm-8.1.0.dist-info

Smartsypants-2.0.1

Introduction to Smartsypants Module

Smartsypants translates plain ASCII punctuation characters into "smart" typographic punctuation HTML entities.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/leohemsted/smartsypants.py/archive/v2.0.1/smartsypants-2.0.1.tar.gz>
- Download MD5 sum: 27957540f4718e892039b2ed208c78f3
- Download size: 24 KB
- Estimated disk space required: 344 KB
- Estimated build time: less than 0.1 SBU

Installation of Smartsypants

Note

The tarball extracts into the `smartsypants.py-2.0.1` directory.

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a working test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user smartsypants
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: smartypants

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/smarty pants-2.0.1.dist-info

Short Descriptions

smartypants translates ASCII punctuation to HTML punctuation.

Snowballstemmer-2.2.0

Introduction to Snowballstemmer Module

The Snowballstemmer package is a small string processing language for creating stemming algorithms for use in Information Retrieval, plus a collection of stemming algorithms implemented using it. Stemming maps different forms of the same word to a common "stem" — for example, the English stemmer maps *connection*, *connections*, *connective*, *connected*, and *connecting* to *connect*. Running a search for *connected* would also find documents which only have the other forms.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/s/snowballstemmer/snowballstemmer-2.2.0.tar.gz>
- Download MD5 sum: 4332ddc7bbe0f344a03915b2ad59a54
- Download size: 85 KB
- Estimated disk space required: 3.8 MB
- Estimated build time: less than 0.1 SBU

Installation of Snowballstemmer

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the *root* user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user snowballstemmer
```

Command Explanations

-w dist: builds the appropriate "wheel" for this module in the directory *dist*.

--no-build-isolation: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

--no-deps: prevents pip3 from building wheels for the project's dependencies.

--no-index: ignores the package index (only looking at --find-links URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory *dist*.

--no-cache-dir: disables the cache to prevent a warning when installing as the *root* user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/snowballstemmer and /usr/lib/python3.12/site-packages/snowballstemmer-2.2.0.dist-info

Sphinxcontrib-applehelp-2.0.0

Introduction to Sphinxcontrib-applehelp Module

The Sphinxcontrib-applehelp package is a Sphinx extension which outputs Apple help books.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/s/sphinxcontrib-applehelp/sphinxcontrib_applehelp-2.0.0.tar.gz
- Download MD5 sum: e16bb1d6199f686d411c180e64a8e831
- Download size: 20 KB
- Estimated disk space required: 1.0 MB (with tests)
- Estimated build time: less than 0.1 SBU

Sphinxcontrib-applehelp Dependencies

Optional (for testing)

[pytest-8.3.2](#) and [sphinx-8.0.2](#) (circular dependency)

Installation of Sphinxcontrib-applehelp

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-applehelp
```

To test the installation, run `pytest`. Several warnings may be issued, due to the use of deprecated functions.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

--no-cache-dir: disables the cache to prevent a warning when installing as the `root` user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_devhelp-2.0.0.dist-info

Sphinxcontrib-devhelp-2.0.0

Introduction to Sphinxcontrib-devhelp Module

The Sphinxcontrib-devhelp package is a Sphinx extension which outputs [Devhelp](#) documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/s/sphinxcontrib-devhelp/sphinxcontrib_devhelp-2.0.0.tar.gz
- Download MD5 sum: 79ef5937b8397f724f4fb065073cd24c
- Download size: 16 KB
- Estimated disk space required: 984 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Sphinxcontrib-devhelp Dependencies

Optional (for testing)

[pytest-8.3.2](#) and [sphinx-8.0.2](#) (circular dependency)

Installation of Sphinxcontrib-devhelp

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-devhelp
```

To test the installation, run `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at --find-links URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory dist.

--no-cache-dir: disables the cache to prevent a warning when installing as the root user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_devhelp-2.0.0.dist-info

Sphinxcontrib-htmlhelp-2.0.1

Introduction to Sphinxcontrib-htmlhelp Module

The Sphinxcontrib-htmlhelp package is a Sphinx extension which renders HTML help files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/s/sphinxcontrib-htmlhelp/sphinxcontrib-htmlhelp-2.0.1.tar.gz>
- Download MD5 sum: e85e6970dc25fbf5fd0f4fa66b897fab
- Download size: 28 KB
- Estimated disk space required: 3.2 MB (add 26 MB for tests)
- Estimated build time: less than 0.1 SBU (with tests)

Sphinxcontrib-htmlhelp Dependencies

Optional (for testing)

[pytest-8.3.2](#), [sphinx-8.0.2](#) (circular dependency), and [html5lib-1.1](#)

Installation of Sphinxcontrib-htmlhelp

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the root user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-htmlhelp
```

If the optional dependencies are installed, the package can be tested with:

```
pytest
```

Command Explanations

-w dist: builds the appropriate "wheel" for this module in the directory dist.

--no-build-isolation: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

--no-deps: prevents pip3 from building wheels for the project's dependencies.

--no-index: ignores the package index (only looking at --find-links URLs instead).

--find-links dist: looks for links to archives such as wheel (.whl) files in the directory dist.

--no-cache-dir: disables the cache to prevent a warning when installing as the root user.

--no-user: Prevent mistakenly running the install command as a non-root user.

--upgrade: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

--force-reinstall: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

--no-deps: Do not install package dependencies. This option may be needed with the --upgrade or --force-reinstall options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_htmlhelp-2.0.1.dist-info

Sphinxcontrib-jquery-4.1

Introduction to Sphinxcontrib-jquery Module

The Sphinxcontrib-jquery package is a Sphinx extension which includes jQuery on newer Sphinx releases.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/s/sphinxcontrib-jquery/sphinxcontrib-jquery-4.1.tar.gz>
- Download MD5 sum: 46ea52845b17343ed6c61e6963fb265d
- Download size: 120 KB
- Estimated disk space required: 552 KB
- Estimated build time: less than 0.1 SBU

Sphinxcontrib-jquery Dependencies

Required

[sphinx-8.0.2](#)

Optional (for testing)

[pytest-8.3.2](#)

Installation of Sphinxcontrib-jquery

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the root user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-jquery
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_jquery-4.1.dist-info

Sphinxcontrib-jsmath-1.0.1

Introduction to Sphinxcontrib-jsmath Module

The Sphinxcontrib-jsmath package is a Sphinx extension which renders display math in HTML via JavaScript.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/s/sphinxcontrib-jsmath/sphinxcontrib-jsmath-1.0.1.tar.gz>
- Download MD5 sum: e45179f0a3608b6766862e0f34c23b62
- Download size: 5.7 KB
- Estimated disk space required: 324 KB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Sphinxcontrib-jsmath Dependencies

Optional (for testing)

[pytest-8.3.2](#) and [sphinx-8.0.2](#) (circular dependency)

Installation of Sphinxcontrib-jsmath

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-jsmath
```

The installation can be tested with the following commands:

```
sed -i 's/text()/read_/' tests/test_jsmath.py &&
pytest
```

the `sed ...` command is needed because of a change in Sphinx API for versions greater than 5.0.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_jsmath-1.0.1.dist-info

Sphinxcontrib-qthelp-2.0.0

Introduction to Sphinxcontrib-qthelp Module

The Sphinxcontrib-qthelp package is a Sphinx extension which outputs QtHelp documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/s/sphinxcontrib-qthelp/sphinxcontrib_qthelp-2.0.0.tar.gz
- Download MD5 sum: ed4f32003b71a54ac3d68aa651cb6573
- Download size: 20 KB
- Estimated disk space required: 18 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Sphinxcontrib-qthelp Dependencies

Optional (for testing)

[pytest-8.3.2](#), [sphinx-8.0.2](#) (circular dependency), and [defusedxml](#)

Installation of Sphinxcontrib-qthelp

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-qthelp
```

The installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&  
testenv/bin/pip3 install defusedxml &&  
testenv/bin/python -m pytest
```

Command Explanations

`--w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_qthelp-2.0.0.dist-info

Sphinxcontrib-serializinghtml-2.0.0

Introduction to Sphinxcontrib-serializinghtml Module

The Sphinxcontrib-serializinghtml package is a Sphinx extension which outputs "serialized" HTML files (json and pickle).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP):
https://files.pythonhosted.org/packages/source/s/sphinxcontrib_serializinghtml/sphinxcontrib_serializinghtml-2.0.0.tar.gz
- Download MD5 sum: b536ce248d5ca134a30018692a17c6ca
- Download size: 16 KB
- Estimated disk space required: 1.1 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Sphinxcontrib-serializinghtml Dependencies

Optional (for testing)

[pytest-8.3.2](#) and [sphinx-8.0.2](#) (circular dependency)

Installation of Sphinxcontrib-serializinghtml

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user sphinxcontrib-serializinghtml
```

To test the installation, first install [sphinx-8.0.2](#), then run `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/sphinxcontrib and /usr/lib/python3.12/site-packages/sphinxcontrib_serializinghtml-2.0.0.dist-info

Trove-Classifiers-2024.7.2

Introduction to Trove-Classifiers Module

Trove-Classifiers is a Python library to encompass all valid PyPI classifiers used to categorize projects and releases per PEP 301, for example Topic :: System :: Filesystems and Development Status :: 6 - Mature.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://files.pythonhosted.org/packages/source/t/trove_classifiers/trove_classifiers-2024.7.2.tar.gz
- Download MD5 sum: ee42acfdbe6fee98be4bcbe0fdaa937c
- Download size: 16 KB
- Estimated disk space required: 276 KB
- Estimated build time: less than 0.1 SBU

Trove-Classifiers Dependencies

Optional (for testing)

[pytest-8.3.2](#)

Installation of Trove-Classifiers

First, hard code the package version into `setup.py` to work around an issue causing the generated wheel contain incorrect version string when the calver module is not installed:

```
sed -i '/calver/s/^/#/; $iversion="2024.7.2"' setup.py
```

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user trove-classifiers
```

To test the installation, issue: `pytest`.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/trove_classifiers and /usr/lib/python3.12/site-packages/trove_classifiers-2024.7.2.dist-info

Typogrify-2.0.7

Introduction to Typogrify Module

Typogrify provides filters to enhance web typography, including support for Django and Jinja templates.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/t/typogrify/typogrify-2.0.7.tar.gz>
- Download MD5 sum: 63f38f80531996f187d2894cc497ba08
- Download size: 13 KB
- Estimated disk space required: 404 KB
- Estimated build time: less than 0.1 SBU

Typogrify Dependencies

Required

[smarty pants-2.0.1](#)

Installation of Typogrify

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

This package does not come with a test suite.

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user typogrify
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/typogrify and /usr/lib/python3.12/site-packages/typogrify-2.0.7.dist-info

Urllib3-2.2.2

Introduction to Urllib3 Module

The Urllib3 module is a powerful, user-friendly HTTP client for Python. It brings many critical features that are missing from the Python standard libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/u/urllib3/urllib3-2.2.2.tar.gz>
- Download MD5 sum: efb79cfdef7b811687afe02ccaf67433
- Download size: 288 KB
- Estimated disk space required: 3.2 MB (add 38 MB for tests)
- Estimated build time: less than 0.1 SBU (1.6 SBU for tests)

Urllib3 Dependencies

Required

[hatchling-1.25.0](#)

Optional (for tests)

[pytest-8.3.2](#), [httpx](#), [hypercorn](#), [mock](#), [PySocks](#), [pytest-timeout](#), [python-dateutil](#), [quart](#), [quart-trio](#), [tornado](#), [trio](#), and [trustme](#)

Installation of Urllib3

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user urllib3
```

Assuming [pytest-8.3.2](#) is installed, but the other optional dependencies are not, the installation can be tested with the following commands:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install trustme \
    tornado \
    python-dateutil \
    mock \
    pysocks \
    pytest-timeout \
    trio \
    hypercom \
    quart \
    quart_trio \
    httpx
python3 /usr/bin/pytest
deactivate
```

A few errors are known to occur.

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/urllib3 and /usr/lib/python3.12/site-packages/urllib3-2.2.2.dist-info

webencodings-0.5.1

Introduction to Webencodings Module

The webencodings module is a Python implementation of the [WHATWG Encoding standard](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.pythonhosted.org/packages/source/w/webencodings/webencodings-0.5.1.tar.gz>
- Download MD5 sum: 32f6e261d52e57bf7e1c4d41546d15b8
- Download size: 12 KB
- Estimated disk space required: 164 KB
- Estimated build time: less than 0.1 SBU

Installation of webencodings

Build the module:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

Now, as the `root` user:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user webencodings
```

Command Explanations

`-w dist`: builds the appropriate "wheel" for this module in the directory `dist`.

`--no-build-isolation`: tells pip3 to run the build in the system environment instead of creating a temporary build environment.

`--no-deps`: prevents pip3 from building wheels for the project's dependencies.

`--no-index`: ignores the package index (only looking at `--find-links` URLs instead).

`--find-links dist`: looks for links to archives such as wheel (.whl) files in the directory `dist`.

`--no-cache-dir`: disables the cache to prevent a warning when installing as the `root` user.

`--no-user`: Prevent mistakenly running the install command as a non-root user.

`--upgrade`: Upgrade the package to the newest available version. This option is used with the install command if a version of the package is already installed.

`--force-reinstall`: Reinstall the package even if it is up-to-date. This option is used with the install command if reinstalling the package or reverting to an earlier version of the package.

`--no-deps`: Do not install package dependencies. This option may be needed with the `--upgrade` or `--force-reinstall` options.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/webencodings and /usr/lib/python3.12/site-packages/webencodings-0.5.1.dist-info

Introduction to Ruby

The Ruby package contains the Ruby development environment. This is useful for object-oriented scripting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://cache.ruby-lang.org/pub/ruby/3.3/ruby-3.3.4.tar.xz>
- Download MD5 sum: e696a878d05867a4d28e62fe9afb7862
- Download size: 16 MB
- Estimated disk space required: 1.4 GB (with C API docs and tests)
- Estimated build time: 2.4 SBU (using parallelism=4; with C API docs; add 0.4 SBU for tests)

Ruby Dependencies

Required

[libyaml-0.2.5](#)

Optional

[Doxygen-1.12.0](#), [Graphviz-12.1.0](#), [rustc-1.80.1](#), [Tk-8.6.14](#), [Valgrind-3.23.0](#), [Berkeley DB](#) (deprecated) and [DTrace](#)

Note

An Internet connection is needed for some tests of this package.

Installation of Ruby

Install Ruby by running the following command:

```
./configure --prefix=/usr      \
            --disable-rpath \
            --enable-shared \
            --without-valgrind \
            --without-baseruby \
            ac_cv_func_qsort_r=no \
            --docdir=/usr/share/doc/ruby-3.3.4 &&
make
```

Optionally, build the CAPI documents by running the following commands:

```
make capi
```

To test the results, issue: `make -j1 -k check`. Note that running the test suite with parallel jobs may cause test failure output as cryptic `unknown object` messages, resulting a summary falsely claiming "all tests passed." If the environment has variables related to proxy settings (`all_proxy`, `ALL_PROXY`, `http_proxy`, and etc.) set, the test suite will bail out early with messages like `net/ftp is not found`. Make sure that you unset these variables for the test suite. There are over 26,000 tests. Some tests related to `ipv6` may indicate errors. If the tests are run in a directory that has a world writable component (e.g. `/tmp`) then several additional tests may fail.

Now, as the `root` user:

```
make install
```

Note

If you have Ruby-On-Rails applications installed and did an upgrade of Ruby by installing this package, you might need to run an update there, too (as the `root` user):

```
cd /path/to/web/app
bundle update rake
```

and restart the webserver which serves the application.

Command Explanations

--disable-rpath: This switch disables embedding `/usr/lib` as a library search path into the `ruby` program. Doing so is not needed (because `/usr/lib` is a system library path) and it may cause the test suite running with the system `libruby.so` instead of the just built one when Ruby has been installed.

--enable-shared: This switch enables building the `libruby` shared library.

--without-baseruby: This switch prevents using the system `ruby` if it is already installed. The build system will use the newly built version instead.

ac_cv_func_qsort_r=no: This switch prevents using the `qsort_r` function from Glibc. Ruby makes some aggressive optimization assuming some implementation details of the sorting algorithm, but the assumption is not true with the Glibc implementation. With this switch Ruby will use its own sort implementation instead.

--disable-install-doc: This switch disables building and installing rdoc indexes and C API documents.

--disable-install-rdoc: This switch disables building and installing rdoc indexes.

--disable-install-capi: This switch disables building and installing C API documents.

Contents

Installed Programs: `bundle`, `bundler`, `erb`, `gem`, `irb`, `racc`, `rake`, `rbs`, `rdbg`, `rdoc`, `ri`, `ruby`, and `typeprof`

Installed Libraries: `libruby.so`

Installed Directories: `/usr/include/ruby-3.3.0`, `/usr/lib/ruby`, `/usr/share/doc/ruby-3.3.4` and `/usr/share/ri`

Short Descriptions

<code>bundle</code>	creates bundles of Ruby Gems
<code>bundler</code>	manages an application's dependencies throughout its lifecycle
<code>erb</code>	is a command line front-end for eRuby, which provides a templating system for Ruby
<code>gem</code>	is the command for RubyGems, which is a sophisticated package manager for Ruby . This is similar to Python's 'pip' command
<code>irb</code>	is the interactive interface for Ruby
<code>rake</code>	is a make-like build utility for Ruby
<code>rdbg</code>	is an interactive debugger for Ruby
<code>rdoc</code>	generates Ruby documentation
<code>ri</code>	displays documentation from a database on Ruby classes, modules, and methods
<code>ruby</code>	is an interpreted scripting language for quick and easy object-oriented programming
<code>libruby.so</code>	contains the API functions required by Ruby

Rustc-1.80.1

Introduction to Rust

The Rust programming language is designed to be a safe, concurrent, practical language.

This package is updated on a six-weekly release cycle. Because it is such a large and slow package to build, is at the moment only required by a few packages in this book, and particularly because newer versions tend to break older mozilla packages, the BLFS editors take the view that it should only be updated when that is necessary (either to fix problems, or to allow a new version of a package to build).

As with many other programming languages, rustc (the rust compiler) needs a binary from which to bootstrap. It will download a stage0 binary at the start of the build, so you cannot compile it without an Internet connection.

Note

Although BLFS usually installs in `/usr`, when you later upgrade to a newer version of rust the old libraries in `/usr/lib/rustlib` will remain, with various hashes in their names, but will not be usable and will waste space. The editors recommend placing the files in the `/opt` directory. In particular, if you have reason to rebuild with a modified configuration (e.g. using the shipped LLVM after building with shared LLVM, perhaps to compile

creates for architectures which the BLFS LLVM build does not support) it is possible for the install to leave a broken `cargo` program. In such a situation, either remove the existing installation first, or use a different prefix such as `/opt/rustc-1.80.1-build2`.

If you prefer, you can of course change the prefix to `/usr`.

The current rustbuild build-system will use all processors, although it does not scale well and often falls back to just using one core while waiting for a library to compile. However it can be mostly limited to a specified number of processors by a combination of adding the switch `--jobs <N>` (e.g. '--jobs 4' to limit to 4 processors) on each invocation of `python3 x.py` and using an environment variable `CARGO_BUILD_JOBS=<N>`. At the moment this is not effective when some of the rustc tests are run.

The current version of rust's num_cpus crate now recognizes that cgroups can be used to restrict which processors it is allowed to use. So if your machine lacks DRAM (typically, less than 2 GB DRAM per core) that might be an alternative to taking CPUs offline. Read [the section called "Use Linux Control Group to Limit the Resource Usage"](#) for how to use a cgroup.

At the moment Rust does not provide any guarantees of a stable ABI.

Note

Rustc defaults to building for ALL supported architectures, using a shipped copy of LLVM. In BLFS the build is only for the X86 architecture. If you intend to develop rust crates, this build may not be good enough for your purposes.

The build times of this version when repeated on the same machine are often reasonably consistent, but as with all compilations using `rustc` there can be some very slow outliers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://static.rust-lang.org/dist/rustc-1.80.1-src.tar.xz>
- Download MD5 sum: 0b00381728b6c005e95194f7e44cff33
- Download size: 191 MB
- Estimated disk space required: 8.9 GB (298 MB installed); add 6.4 GB if running the tests
- Estimated build time: 6.7 SBU (including download time; add 6.2 SBU for tests, both using parallelism=8)

Rust Dependencies

Required

[CMake-3.30.2](#) and [cURL-8.9.1](#)

Note

An Internet connection is needed for building this package.

Recommended

[libssh2-1.11.0](#), [LLVM-18.1.7](#) (built with -D LLVM_LINK_LLVM_DYLIB=ON so that rust can link to system LLVM instead of building its shipped version), and [SQLite-3.46.1](#)

Note

If a recommended dependency is not installed, a shipped copy in the Rustc source tarball will be built and used.

Optional

[GDB-15.1](#) (used by the test suite if it is present), [git-2.46.0](#) (required by the test suite), [cranelift](#), [jemalloc](#), libgccjit (read command explanation in [GCC-14.2.0](#)), and [libgit2](#)

Installation of Rust

To install into the `/opt` directory, remove any existing `/opt/rustc` symlink and create a new directory (i.e. with a different name if trying a modified build of the same version). As the `root` user:

```
mkdir -pv /opt/rustc-1.80.1      &&
ln -svfn rustc-1.80.1 /opt/rustc
```

Note

If multiple versions of Rust are installed in `/opt`, changing to another version only requires changing the `/opt/rustc` symbolic link and then running `ldconfig`.

Create a suitable `config.toml` file which will configure the build.

```
cat << EOF > config.toml
# see config.toml.example for more possible options
# See the 8.4 book for an old example using shipped LLVM
# e.g. if not installing clang, or using a version before 13.0

# Tell x.py the editors have reviewed the content of this file
# and updated it to follow the major changes of the building system,
# so x.py will not warn us to do such a review.
change-id = 125535

[llvm]
# by default, rust will build for a myriad of architectures
targets = "x86"

# When using system llvm prefer shared libraries
link-shared = true

[build]
# omit docs to save time and space (default is to build them)
docs = false

# install extended tools: cargo, clippy, etc
extended = true

# Do not query new versions of dependencies online.
locked-deps = true

# Specify which extended tools (those from the default install).
tools = ["cargo", "clippy", "rustdoc", "rustfmt"]

# Use the source code shipped in the tarball for the dependencies.
# The combination of this and the "locked-deps" entry avoids downloading
# many crates from Internet, and makes the Rustc build more stable.
vendor = true

[install]
prefix = "/opt/rustc-1.80.1"
docdir = "share/doc/rustc-1.80.1"

[rust]
channel = "stable"
description = "for BLFS 12.2"

# Enable the same optimizations as the official upstream build.
lto = "thin"
codegen-units = 1

[target.x86_64-unknown-linux-gnu]
# NB the output of llvm-config (i.e. help options) may be
# dumped to the screen when config.toml is parsed.
llvm-config = "/usr/bin/llvm-config"

[target.i686-unknown-linux-gnu]
# NB the output of llvm-config (i.e. help options) may be
```

```
# dumped to the screen when config.toml is parsed.  
llvm-config = "/usr/bin/llvm-config"  
EOF
```

Note

The `python3 x.py` commands may output a warning message complaining no codegen-backends config matched the requested path to build a codegen backend. And the provided "suggestion" (add backend to codegen-backends in `config.toml`) will not silence it. This warning is [bogus](#) and it should be ignored.

Compile Rust by running the following commands:

```
{ [ ! -e /usr/include/libssh2.h ] ||  
  export LIBSSH2_SYS_USE_PKG_CONFIG=1; } &&  
{ [ ! -e /usr/include/sqlite3.h ] ||  
  export LIBSQLITE3_SYS_USE_PKG_CONFIG=1; } &&  
python3 x.py build
```

Note

The test suite will generate some messages in the [systemd journal](#) for traps on invalid opcodes, and for segmentation faults. In themselves these are nothing to worry about, just a way for the test to be terminated.

To run the tests (again using all available CPUs) issue:

```
SSL_CERT_DIR=/etc/ssl/certs \  
python3 x.py test --verbose --no-fail-fast --keep-stage-std=1 | \  
tee rustc-testlog
```

Three tests named `tests/run-make/print-cfg`, `tests/run-make/print-to-output`, and `core::config::tests::download_ci_llvm` are known to fail.

As with all large test suites, some tests might fail on some machines - if the number of additional failures is low, check the log for 'failures:' and review lines above that, particularly the 'stderr:' lines. Any mention of SIGSEGV or signal 11 in a failing test is a cause for concern.

If you get any *other* failing test which reports an issue number then you should search for that issue. For example, when `rustc >= 1.41.1` was built with a version of sysllvm before 10.0 the test for issue 69225 failed <https://github.com/rust-lang/rust/issues/69225> and that should be regarded as a critical failure (they released 1.41.1 because of it). Most other failures will not be critical.

Therefore, you should determine the number of failures. The number of tests which passed and failed can be found by running:

```
grep '^test result:' rustc-testlog |  
awk '{sum1 += $4; sum2 += $6} END { print sum1 " passed; " sum2 " failed" }'
```

The other available fields are \$8 for those which were ignored (i.e. skipped), \$10 for 'measured' and \$12 for 'filtered out' but both those last two are probably zero.

Now, as the `root` user, install the package:

Note

If `sudo` or `su` is invoked for switching to the `root` user, ensure `LIBSSH2_SYS_USE_PKG_CONFIG` and `LIBSQLITE3_SYS_USE_PKG_CONFIG` are correctly passed or the following command may rebuild `cargo` with shipped copies of libssh2 and sqlite. For `sudo`, use the `--preserve-env=LIB{SSH2,SQLITE3}_SYS_USE_PKG_CONFIG` option. For `su`, do not use the `-` or `--login` options.

```
python3 x.py install rustc std &&  
install -vM755 \  
build/host/stage1-tools/**/{cargo{,-clippy,-fmt},clippy-driver,rustfmt} \  
/opt/rustc-1.80.1/bin &&  
install -vDm644 \  
src/tools/cargo/src/etc/_cargo \  
/etc/cargo
```

```
/opt/rustc-1.80.1/share/zsh/site-functions/_cargo &&
install -vm644 src/tools/cargo/src/etc/man/* \
/opt/rustc-1.80.1/share/man/man1
```

Still as the `root` user, fix the installation of documentation and symlink a Zsh completion file into the correct location:

```
rm -fv /opt/rustc-1.80.1/share/doc/rustc-1.80.1/*.old &&
install -vm644 README.md \
/opt/rustc-1.80.1/share/doc/rustc-1.80.1 &&

install -vdm755 /usr/share/zsh/site-functions &&
ln -sfv /opt/rustc/share/zsh/site-functions/_cargo \
/usr/share/zsh/site-functions
```

Finally, unset the exported environment variables:

```
unset LIB{SSH2,SQLITE3}_SYS_USE_PKG_CONFIG
```

Command Explanations

`ln -svfn rustc-1.80.1 /opt/rustc`: if this is not the first use of the `/opt/rustc` symlink, overwrite it by forcing, and use the '`-n`' flag to avoid getting confusing results from e.g. `ls -l`.

`targets = "x86"`: this avoids building all the available linux cross-compilers (AArch64, MIPS, PowerPC, SystemZ, etc). Unfortunately, rust insists on installing source files for these below `/opt/rustc/lib/src`.

`extended = true`: this installs several tools (specified by the `tools` entry) alongside `rustc`.

`tools = ["cargo", "clippy", "rustdoc", "rustfmt"]`: only build the tools from the 'default' profile in binary command `rustup` which are recommended for most users. The other tools are unlikely to be useful unless using (old) code analyzers or editing the standard library.

`channel = "stable"`: this ensures only stable features can be used, the default in `config.toml` is to use development features, which is not appropriate for a released version.

`[target.x86_64-unknown-linux-gnu]`: the syntax of `config.toml` requires an `llvm-config` entry for each target for which system-`llvm` is to be used. Change the target to `[target.i686-unknown-linux-gnu]` if you are building on 32-bit x86. This whole section may be omitted if you wish to build against the shipped `llvm`, or do not have `clang`, but the resulting build will be larger and take longer.

`export LIBSSH2_SYS_USE_PKG_CONFIG=1`: Allow `cargo` to link to system libssh2.

`export LIBSQLITE3_SYS_USE_PKG_CONFIG=1`: Allow `cargo` to link to system sqlite.

`SSL_CERT_DIR=/etc/ssl/certs`: Work around an issue causing test failures with the CA certificate store layout used by [make-ca-1.14](#).

`--verbose`: this switch can sometimes provide more information about a test which fails.

`--no-fail-fast`: this switch ensures that the test suite will not stop at the first error.

`--keep-stage-std 1`: this switch prevents `x.py test` from rebuilding the entire package for some unknown reason.

`install ... /opt/rustc-1.80.1/...`: these commands explicitly install some Rust components. They are used instead of relying on `x.py install` because the latter would rebuild these components for some unknown reason.

Configuring Rust

Configuration Information

If you installed `rustc` in `/opt`, you need to update the following configuration files so that `rustc` is correctly found by other packages and system processes.

As the `root` user, create the `/etc/profile.d/rustc.sh` file:

```
cat > /etc/profile.d/rustc.sh << "EOF"
# Begin /etc/profile.d/rustc.sh

pathprepend /opt/rustc/bin          PATH
```

```
# End /etc/profile.d/rustc.sh  
EOF
```

Immediately after installation, update the current PATH for your current shell as a normal user:

```
source /etc/profile.d/rustc.sh
```

Contents

Installed Programs: cargo-clippy, cargo-fmt, cargo, clippy-driver, rust-gdb, rust-gdbgui, rust-lldb, rustc, rustdoc, and rustfmt

Installed Libraries: librustc-driver-<16-byte-hash>.so, libstd-<16-byte-hash>.so, and libtest-<16-byte-hash>.so

Installed Directories: ~/cargo, /opt/rustc, symbolic link to /opt/rustc-1.80.1

Short Descriptions

cargo-clippy	provides lint checks for a cargo package
cargo-fmt	formats all bin and lib files of the current crate using rustfmt
cargo	is the Package Manager for Rust
clippy-driver	provides lint checks for Rust
rust-gdb	is a wrapper script for gdb, pulling in Python pretty-printing modules installed in /opt/rustc-1.80.1/lib/rustlib/etc
rust-gdbgui	is a wrapper script for a graphical front end to gdb that runs in a browser
rust-lldb	is a wrapper script for LLDB (the LLVM debugger) pulling in the Python pretty-printing modules
rustc	is the rust compiler
rustdoc	generates documentation from rust source code
rustfmt	formats rust code
libstd-<16-byte-hash>.so	is the Rust Standard Library, the foundation of portable Rust software

rust-bindgen-0.70.0

Introduction to rust-bindgen

The rust-bindgen package contains a utility that generates Rust bindings from C/C++ headers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/rust-lang/rust-bindgen/archive/v0.70.0/rust-bindgen-0.70.0.tar.gz>
- Download MD5 sum: 05dfba889299f3b666c43bbd1256af9f
- Download size: 2.3 MB
- Estimated disk space required: 178 MB
- Estimated build time: 0.5 SBU (with parallelism=8)

rust-bindgen Dependencies

Required

[rustc-1.80.1](#) and [LLVM-18.1.7](#) (with Clang, runtime)

Note

An Internet connection is needed for building this package.

Installation of rust-bindgen

Install rust-bindgen by running the following commands:

```
cargo build --release
```

To test the results, issue: `cargo test --release`.

Now, as the `root` user:

```
install -v -m755 target/release/bindgen /usr/bin
```

Contents

Installed Programs: bindgen

Installed Libraries: None

Installed Directories: None

Short Descriptions

`bindgen` generates bindings for Rust from C/C++ headers

SCons-4.8.0

Introduction to SCons

SCons is a tool for building software (and other files) implemented in Python.

Note

When expanding the tarball the package expands to scons-4.8.0, not the expected SCons-4.8.0.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/scons/SCons-4.8.0.tar.gz>
- Download MD5 sum: 68a13ef0d3515044544f8bd0949befa2
- Download size: 3.1 MB
- Estimated disk space required: 50 MB
- Estimated build time: less than 0.1 SBU

Installation of SCons

Build the package:

```
pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
```

As the `root` user, install the package:

```
pip3 install --no-index --find-links=dist --no-cache-dir --no-user SCons &&  
install -v -m644 *.1 /usr/share/man/man1
```

Contents

Installed Programs: scons, scons-configure-cache, and sconsign

Installed Libraries: None

Installed Directory: /usr/lib/python3.12/site-packages/SCons{-4.8.0.dist-info}

Short Descriptions

<code>scons</code>	is a software construction tool
<code>scons-configure-cache</code>	shows or converts the configuration of an SCons cache directory
<code>sconsign</code>	prints SCons .sconsign file information

slang-2.3.3

Introduction to slang

S-Lang (slang) is an interpreted language that may be embedded into an application to make the application extensible. It provides facilities required by interactive applications such as display/screen management, keyboard input and keymaps.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.jedsoft.org/releases/slanguage/slanguage-2.3.3.tar.bz2>
- Download MD5 sum: 69015c8300088373eb65ffcc6ed4db8c
- Download size: 1.6 MB
- Estimated disk space required: 22 MB (add 15 MB for tests)
- Estimated build time: 0.4 SBU (add 0.5 SBU for tests)

Slang Dependencies

Optional

[libpng-1.6.43](#) and [Oniguruma](#)

Installation of Slang

Note

This package does not support parallel build.

Install slang by running the following commands:

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --with-readline=gnu &&
make -j1 RPATH=
```

To test the results, issue: `LC_ALL=C make check`.

Now, as the `root` user:

```
make install_doc_dir=/usr/share/doc/slanguange-2.3.3 \
SLSH_DOC_DIR=/usr/share/doc/slanguange-2.3.3/sls \
RPATH= install
```

Command Explanations

`--with-readline=gnu`: This parameter sets GNU Readline to be used by the parser interface instead of the slang internal version.

`RPATH=`: This overridden `make` variable prevents hard coding library search paths (rpath) into the binary executable files and shared libraries. This package does not need rpath for an installation into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`install_doc_dir=/usr/share/doc/slanguange-2.3.3 SLSH_DOC_DIR=/usr/share/doc/slanguange-2.3.3/sls`: These overridden `make` variables ensure installing this package with a versioned documentation installation directory.

Configuring slang

Config Files

`~/.slshrc` and `/etc/slsh.rc`

Contents

Installed Program: slsh

Installed Libraries: libslang.so and numerous support modules

Installed Directories: /usr/lib/slang, /usr/share/doc/slang-2.3.3 and /usr/share/slsh

Short Descriptions

slsh is a simple program for interpreting slang scripts. It supports dynamic loading of slang modules and includes a Readline interface for interactive use

Subversion-1.14.3

Introduction to Subversion

Subversion is a version control system that is designed to be a compelling replacement for CVS in the open source community. It extends and enhances CVS' feature set, while maintaining a similar interface for those already familiar with CVS. These instructions install the client and server software used to manipulate a Subversion repository. Creation of a repository is covered at [Running a Subversion Server](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist/subversion/subversion-1.14.3.tar.bz2>
- Download MD5 sum: 19756a5ceb32a022698a66e48616ef6b
- Download size: 8.2 MB
- Estimated disk space required: 193 MB (add 189 MB for bindings, 54 MB for docs, 1.3 GB for tests)
- Estimated build time: 0.5 SBU (Using parallelism=4; add 2.0 SBU for bindings, 30 SBU for tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/subversion-1.14.3-upstream_fixes-1.patch

Subversion Dependencies

Required

[Apr-Util-1.6.3](#) and [SQLite-3.46.1](#)

Recommended

[Serf-1.3.10](#) (for handling http:// and https:// URLs)

Optional

[Apache-2.4.62](#), [Boost-1.86.0](#), [Cyrus SASL-2.1.28](#), [dbus-1.14.10](#), [Doxygen-1.12.0](#) (for generating HTML documentation), [gnome-keyring-46.2](#), [libsecret-0.21.4](#), [Py3c-1.4](#) (for the python bindings, and tests), [Python-3.12.5](#) (with sqlite support for the tests), [Ruby-3.3.4](#), [SWIG-4.2.1](#) (for building Perl, Python and Ruby bindings), and [UTF8proc](#)

Optional (for the Java Bindings)

One of [OpenJDK-22.0.2](#), [Dante](#) or [Jikes](#), [JUnit 4](#) (to test the Java bindings) and [apache-ant-1.10.14](#).

Installation of Subversion

First, adapt some Python scripts to use python3:

```
grep -rl '^#!.*python$' | xargs sed -i '1s/python/&3/'
```

Next, fix building this package with GCC 14:

```
patch -Np1 -i ../subversion-1.14.3-upstream_fixes-1.patch
```

After modifying the SWIG bindings like we did with the patch, the build system must be regenerated so the bindings pick up the changes:

```
touch build/generator/swig/*.py &&
mv -v build-outputs.mk{,.old} &&
./autogen.sh --release
```

Install Subversion by running the following commands:

```
PYTHON=python3 \
./configure --prefix=/usr \
--disable-static \
--with-apache-libexecdir \
--with-utf8proc=internal &&
make
```

If you have [Doxygen-1.12.0](#) installed and you wish to build the API documentation, issue:

```
doxygen doc/doxygen.conf
```

If you wish to build the Java bindings pass the `--enable-javahl` parameter to the `configure` command. In addition, if you want to run the Java test suite, you have to specify the location of the JUnit file by adding `--with-junit=<path to junit jar>` (for instance `--with-junit=/usr/local/java/lib/junit-4.13.jar`) to `configure`. The JUnit jar file is no longer included in [apache-ant-1.10.14](#) and must be downloaded separately. To build the Java bindings, issue the following command:

```
make -j1 javahl
```

If you want to compile Perl, Python, or Ruby bindings, issue any of the following command:

```
make swig-pl # for Perl
make swig-py \
  swig_pydir=/usr/lib/python3.12/site-packages/libsvn \
  swig_pydir_extra=/usr/lib/python3.12/site-packages/svn # for Python
make swig-rb # for Ruby
```

To test the results, issue: `make check`. Four tests in the `commit_tests.py`, `prop_tests.py`, and `update_tests.py` suites are known to fail. Due to changes in Python-3.12, you can expect to see around 200 Syntax Warnings during the test suite due to invalid escape sequences.

To test the results of any of the SWIG bindings, you can use any of the following commands: `make check-swig-pl`, `make check-swig-py`, OR `make check-swig-rb`.

Now, as the `root` user:

```
make install &&
install -v -m755 -d /usr/share/doc/subversion-1.14.3 &&
cp      -v -R doc/* /usr/share/doc/subversion-1.14.3
```

If you built the Java bindings, issue the following command as the `root` user to install them:

```
make install-javahl
```

If you built the Perl, Python, or Ruby bindings, issue any of the following commands as the `root` user to install them:

```
make install-swig-pl
make install-swig-py \
  swig_pydir=/usr/lib/python3.12/site-packages/libsvn \
  swig_pydir_extra=/usr/lib/python3.12/site-packages/svn
make install-swig-rb
```

The java bindings need to be installed in order for the tests to run, since the tests attempt to look for them in CLASSPATH. To test the results of the Java bindings build, issue `LANG=C make check-javahl`.

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-apache-libexecdir`: If [Apache-2.4.62](#) is installed, the shared Apache modules are built. This switch allows to have those modules installed to Apache's configured module dir instead of `/usr/libexec`. It has no effect if Apache is not installed.

`--with-utf8proc=internal`: Remove these switches if you have installed the optional dependencies.

```
--enable-javahl: enables compilation of Java high level bindings. Running make javahl is necessary to do the compilation.  
--with-junit=<location of the junit jar file>: gives the location of the junit jar, otherwise the javahl tests cannot be run.  
--disable-gmock: Do not use the Googlemock testing framework.
```

Configuring Subversion

Config Files

`~/.subversion/config` and `/etc/subversion/config`

Configuration Information

`/etc/subversion/config` is the Subversion system-wide configuration file. This file is used to specify defaults for different `svn` commands.

`~/.subversion/config` is the user's personal configuration file. It is used to override the system-wide defaults set in `/etc/subversion/config`.

Contents

Installed Programs:	svn, svnadmin, svnbench, svndumpfilter, svnfsfs, svnlook, svnmucc, svnrdump, svnserve, svnsync, and svnversion
Installed Libraries:	libsvn_*-1.so and optionally, a Java library, the mod_dav_svn.so and mod_authz_svn.so Apache HTTP DSO modules and various Perl, Python and Ruby modules.
Installed Directories:	/usr/include/subversion-1, /usr/lib/perl5/site_perl/5.40/{,auto/}SVN, /usr/lib/python3.12/site-packages/{libsvn,svn}, /usr/lib/ruby/site_ruby/<x.y.z>/{'<arch-linux>-linux/}svn, /usr/lib/svn-javahl, and /usr/share/doc/subversion-1.14.3

Short Descriptions

<code>svn</code>	is a command-line client program used to access Subversion repositories
<code>svnadmin</code>	is a tool for creating, tweaking or repairing a Subversion repository
<code>svnbench</code>	is a benchmarking tool
<code>svndumpfilter</code>	is a program for filtering Subversion repository dumpfile format streams
<code>svnfsfs</code>	is the FSFS (FileSystem atop of the FileSystem - Subversion filesystem implementation) repository manipulation tool
<code>svnlook</code>	is a tool for inspecting a Subversion repository
<code>svnmucc</code>	is a Multiple URL Command Client for Subversion
<code>svnrdump</code>	is a tool for dumping or loading a remote Subversion repository
<code>svnserve</code>	is a custom standalone server program, able to run as a daemon process or invoked by SSH
<code>svnsync</code>	is a Subversion repository synchronisation tool
<code>svnversion</code>	is used to report the version number and state of a working Subversion repository copy
<code>libsvn_*-1.so</code>	are the support libraries used by the Subversion programs
<code>mod_authz_svn.so</code>	is a plug-in module for the Apache HTTP server, used to authenticate users to a Subversion repository over the Internet or an intranet
<code>mod_dav_svn.so</code>	is a plug-in module for the Apache HTTP server, used to make a Subversion repository available to others over the Internet or an intranet

Running a Subversion Server

Running a Subversion Server

This section will describe how to set up, administer and secure a Subversion server.

Subversion Server Dependencies

Required

[Subversion-1.14.3](#) and [OpenSSH-9.8p1](#)

Setting up a Subversion Server.

The following instructions will install a Subversion server, which will be set up to use OpenSSH as the secure remote access method, with `svnserve` available for anonymous access.

Configuration of the Subversion server consists of the following steps:

1. Set Up Users, Groups, and Permissions

You'll need to be user `root` for the initial portion of configuration. Create the `svn` user and group with the following commands:

```
groupadd -g 56 svn &&
useradd -c "SVN Owner" -d /home/svn -m -g svn -s /bin/false -u 56 svn
```

If you plan to have multiple repositories, you should have a group dedicated to each repository for ease of administration. Create the `svntest` group for the test repository and add the `svn` user to that group with the following commands:

```
groupadd -g 57 svntest &&
usermod -G svntest -a svn
```

Additionally you should set `umask 002` while working with a repository so that all new files will be writable by owner and group. This is made mandatory by creating a wrapper script for `svn` and `svnserve`:

```
mv /usr/bin/svn /usr/bin/svn.orig &&
mv /usr/bin/svnservice /usr/bin/svnservice.orig &&
cat >> /usr/bin/svn << "EOF"
#!/bin/sh
umask 002
/usr/bin/svn.orig "$@"
EOF
cat >> /usr/bin/svnservice << "EOF"
#!/bin/sh
umask 002
/usr/bin/svnservice.orig "$@"
EOF
chmod 0755 /usr/bin/svn{,serve}
```

Note

If you use Apache for working with the repository over HTTP, even for anonymous access, you should wrap `/usr/sbin/httpd` in a similar script.

2. Create a Subversion repository.

There are several ways to set up a subversion repository. It is recommended to have a look at the [SVN Book](#) corresponding chapter. A basic repository can be set up with the instructions below.

Create a new Subversion repository with the following commands (as the `root` user):

```
install -v -m 0755 -d /srv/svn &&
install -v -m 0755 -o svn -g svn -d /srv/svn/repositories &&
svnadmin create /srv/svn/repositories/svntest
```

Now that the repository is created, it should be populated with something useful. You'll need to have a predefined directory layout set up exactly as you want your repository to look. For example, here is a sample BLFS layout setup with a root of `svntest/`. You'll need to set up a directory tree similar to the following:

```
svntest/          # The name of the repository
  trunk/         # Contains the existing source tree
    BOOK/
    bootscripts/
    edguide/
    patches/
    scripts/
  branches/      # Needed for additional branches
  tags/          # Needed for tagging release points
```

Once you've created your directory layout as shown above, you are ready to do the initial import:

```
svn import -m "Initial import." \
</path/to/source/tree> \
file:///srv/svn/repositories/svntest
```

Now change owner and group information on the repository, and add an unprivileged user to the `svn` and `svntest` groups:

```
chown -R svn:svntest /srv/svn/repositories/svntest    &&
chmod -R g+w      /srv/svn/repositories/svntest    &&
chmod g+s       /srv/svn/repositories/svntest/db &&
usermod -G svn,svntest -a <username>
```

`svntest` is the group assigned to the `svntest` repository. As mentioned earlier, this eases administration of multiple repositories when using OpenSSH for authentication. Going forward, you'll need to add your unprivileged user, and any additional users that you wish to have write access to the repository, to the `svn` and `svntest` groups.

In addition, you'll notice that the new repository's `db` directory is set-groupID. If the reasoning is not immediately obvious, when using any external authentication method (such as `ssh`), the sticky bit is set so that all new files will be owned by the user, but group of `svntest`. Anyone in the `svntest` group can create files, but still give the entire group write access to those files. This avoids locking out other users from the repository.

Now, return to an unprivileged user account, and take a look at the new repository using `svnlook`:

```
svnlook tree /srv/svn/repositories/svntest/
```

Note

You may need to log out and back in again to refresh your group memberships. `su <username>` should work as well.

3. Configure the Server

As mentioned previously, these instructions will configure the server to use only `ssh` for write access to the repository and to provide anonymous access using `svnserve`. There are several other ways to provide access to the repository. These additional configurations are best explained at <https://svnbook.red-bean.com/>.

Access configuration needs to be done for each repository. Create the `svnserve.conf` file for the `svntest` repository using the following commands:

```
cp /srv/svn/repositories/svntest/conf/svnserve.conf \
/srv/svn/repositories/svntest/conf/svnserve.conf.default &&

cat > /srv/svn/repositories/svntest/conf/svnserve.conf << "EOF"
[general]
anon-access = read
auth-access = write
EOF
```

There is not a lot to the configuration file at all. You'll notice that only the general section is required. Take a look at the `svnserve.conf.default` file for information on using `svnserve`'s built-in authentication method.

4. Starting the Server

To start the server at boot time, install the `svnserve.service` unit from the [blfs-systemd-units-20240801](#) package:

```
make install-svnserve
```

Additionally, the instructions above require that svn server uses `umask 002` so that all new files will be writable by owner and group. This can be achieved by creating a systemd unit override file by running the following command:

```
mkdir -p /etc/systemd/system
echo "UMask=0002" > /etc/systemd/system/svnserve.service.d/99-user.conf
```

Options which are passed to `svnserve` daemon can be changed in `/etc/default/svnserve`.

Introduction to SWIG

SWIG (Simplified Wrapper and Interface Generator) is a compiler that integrates C and C++ with languages including Perl, Python, Tcl, Ruby, PHP, Java, JavaScript, C#, D, Go, Lua, Octave, R, Racket, Scilab, Scheme, and Ocaml. SWIG can also export its parse tree into Lisp s-expressions and XML.

SWIG reads annotated C/C++ header files and creates wrapper code (glue code) in order to make the corresponding C/C++ libraries available to the listed languages, or to extend C/C++ programs with a scripting language.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/swig/swig-4.2.1.tar.gz>
- Download MD5 sum: 7697b443d7845381d64c90ab54d244af
- Download size: 8.0 MB
- Estimated disk space required: 81 MB (1.8 GB with tests)
- Estimated build time: 0.1 SBU (add 7.7 SBU for tests; both using parallelism=4)

SWIG Dependencies

Required

[pcre2-10.44](#)

Optional

[Boost-1.86.0](#) for tests, and any of the languages mentioned in the introduction, as run-time dependencies

Installation of SWIG

Install SWIG by running the following commands:

```
./configure --prefix=/usr \
            \
            \
            \
make
```

To test the results, issue: `PY3=1 make TCL_INCLUDE= -k check`. The unsetting of the variable `TCL_INCLUDE` is necessary since it is not correctly set by `configure`. The tests are only executed for the languages installed on your machine, so the disk space and SBU values given for the tests may vary, and should be considered as mere orders of magnitude. According to SWIG's documentation, the failure of some tests should not be considered harmful. The go tests are buggy and may generate a lot of meaningless output.

Now, as the `root` user:

```
make install &&
cp -v -R Doc -T /usr/share/doc/swig-4.2.1
```

Command Explanations

`--without-maximum-compile-warnings`: disables compiler ansi conformance enforcement, which triggers errors in the Lua headers (starting with Lua 5.3).

`--without-<language>`: allows disabling the building of tests and examples for <language>, but all the languages capabilities of SWIG are always built. This switch is used for JavaScript because the SWIG implementation is incomplete and a lot of tests fail due to API changes in Node-20.

Contents

Installed Programs: swig and ccache-swig

Installed Library: None

Installed Directories: /usr/share/doc/swig-4.2.1 and /usr/share/swig

Short Descriptions

`swig` takes an interface file containing C/C++ declarations and SWIG special instructions, and generates the corresponding wrapper code needed to build extension modules

`ccache-` is a compiler cache, which speeds up re-compilation of C/C++/SWIG code
`swig`

Tk-8.6.14

Introduction to Tk

The Tk package contains a TCL GUI Toolkit.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/tcl/tk8.6.14-src.tar.gz>
- Download MD5 sum: cf2aaac0478ef468b48e65c10e6b0d07
- Download size: 4.3 MB
- Estimated disk space required: 25 MB
- Estimated build time: 0.3 SBU (add 2.0 SBU for tests)

Tk Dependencies

Required

[Xorg Libraries](#)

Installation of Tk

Install Tk by running the following commands:

```
cd unix &&
./configure --prefix=/usr \
    --mandir=/usr/share/man \
    ${([ $uname -m ] = x86_64 ] && echo --enable-64bit) &&

make &&

sed -e "s@^\\(TK_SRC_DIR=''.*)@\\1/usr/include@" \
    -e "/TK_B/s@'=\\(-L\\)\\?.*unix@='\\1/usr/lib@" \
    -i tkConfig.sh
```

Running the tests is not recommended. Failures will be reported during the tests, depending on the screen resolution/capabilities, fonts installed and other X related parameters, but the end report can show 0 failures. Some tests will steal focus and some might crash your X Server. To test the results anyway, issue: `make test`. Ensure you run it from an X Window display device with the GLX extensions loaded, but even so, tests might hang.

Now, as the `root` user:

```
make install &&
make install-private-headers &&
ln -v -sf wish8.6 /usr/bin/wish &&
chmod -v 755 /usr/lib/libtk8.6.so
```

Command Explanations

`--enable-64bit`: This switch is used to enable 64 bit support in Tk on 64 bit operating systems.

`make install-private-headers`: This command is used to install the Tk library interface headers used by other packages if they link to the Tk library.

`ln -v -sf wish8.6 /usr/bin/wish`: This command is used to create a compatibility symbolic link to the `wish8.6` file as many packages expect a file named `wish`.

`sed -e ... tkConfig.sh`: The Tk package expects that its source tree is preserved so that packages depending on it for their compilation can utilize it. This `sed` removes the references to the build directory and replaces them with saner system-wide locations.

Contents

Installed Programs: wish and wish8.6

Installed Libraries: libtk8.6.so and libtkstub8.6.a

Installed Directory: /usr/lib/tk8.6

Short Descriptions

wish	is a symlink to the <code>wish8.6</code> program
wish8.6	is a simple shell containing the Tk toolkit that creates a main window and then processes Tcl commands
libtk8.6.so	contains the API functions required by Tk

unifdef-2.12

Introduction to unifdef

The unifdef package contains a utility that is useful for removing preprocessor conditionals from code.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://dotat.at/prog/unifdef/unifdef-2.12.tar.gz>
- Download MD5 sum: b225312c110cd2600ca7166bd0419751
- Download size: 88 KB
- Estimated disk space required: 1.9 MB
- Estimated build time: less than 0.1 SBU

Installation of unifdef

Install unifdef by running the following commands:

```
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make prefix=/usr install
```

Contents

Installed Programs: unifdef and unifdefall

Installed Libraries: None

Installed Directories: None

Short Descriptions

unifdef	removes preprocessor conditionals from code
unifdefall	removes preprocessor conditionals from code across a whole tree

Vala-0.56.17

Introduction to Vala

Vala is a new programming language that aims to bring modern programming language features to GNOME developers without imposing any additional runtime requirements and without using a different ABI compared to applications and libraries written in C.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/vala/0.56/vala-0.56.17.tar.xz>
- Download MD5 sum: 134075855867fdd9c51ca7555c4951bb
- Download size: 3.8 MB
- Estimated disk space required: 160 MB (add 19 MB for tests)
- Estimated build time: 0.5 SBU (add 1.0 SBU for tests; both using parallelism=4)

Vala Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection required for the tests)

Recommended

[Graphviz-12.1.0](#) (Required for valadoc)

Optional

[dbus-1.14.10](#) (Required for the tests), [libxslt-1.1.42](#) (Required for generating the documentation), [help2man](#), [jing](#), and [weasyprint](#)

Installation of Vala

Install Vala by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

bootstrap: This `make` target forces the building system to build the package twice and use the `vala` executable produced in the first build to regenerate the `.c` files from the `.vala` files if needed. If a `vala` executable is not installed yet and you've modified any `.vala` files in the source tree, run `make bootstrap` instead of `make`.

--disable-valadoc: This option is required if [Graphviz-12.1.0](#) is not installed.

Contents

Installed Programs:	vala, vala-0.56, valac, valadoc, vala-gen-introspect, and vapigen (symlinks); valac-0.56, valadoc-0.56, vala-gen-introspect-0.56, and vapigen-0.56
Installed Library:	libvala-0.56.so and libvaladoc-0.56.so
Installed Directories:	/usr/include/vala-0.56, /usr/include/valadoc-0.56, /usr/lib/vala-0.56, /usr/lib/valadoc-0.56, /usr/share/vala, /usr/share/vala-0.56, /usr/share/valadoc-0.56, and /usr/share/devhelp/books/vala-0.56

Short Descriptions

<code>valac</code>	is a compiler that translates Vala source code into C source and header files
<code>valadoc</code>	is a documentation generator for generating API documentation from Vala source code based on libvala
<code>vala-gen-introspect</code>	generates a GI file for GObject and GLib based packages
<code>vapigen</code>	is an utility which generates Vala API (VAPI) files from GI files
<code>libvala-0.56.so</code>	contains the Vala API functions

Introduction to Valgrind

Valgrind is an instrumentation framework for building dynamic analysis tools. There are Valgrind tools that can automatically detect many memory management and threading bugs, and profile programs in detail. Valgrind can also be used to build new tools.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sourceware.org/pub/valgrind/valgrind-3.23.0.tar.bz2>
- Download MD5 sum: c59775fcfa82fcce796843d0aaa7538
- Download size: 16 MB
- Estimated disk space required: 392 MB (add 68 MB for tests)
- Estimated build time: 0.5 SBU (add 7.2 SBU for tests; both using parallelism=4)

Valgrind Dependencies

Optional

[docbook-xmL-4.5](#) (for tests), [GDB-15.1](#) (for tests), [LLVM-18.1.7](#) (with Clang), and [Which-2.21](#) (for tests)

Installation of Valgrind

Install Valgrind by running the following commands:

```
sed -i 's|/doc/valgrind||' docs/Makefile.in &&
./configure --prefix=/usr \
             --datadir=/usr/share/doc/valgrind-3.23.0 &&
make
```

To test the results, issue: `make regtest`. The tests may hang forever if [GDB-15.1](#) is not installed. Some tests are known to hang also, depending on the version of glibc. A few tests can fail in various suites. Problematic tests can be disabled by changing the `prereq:` line in the corresponding `.vgtest` file to `prereq: false`. For example:

```
sed -e 's@prereq:@prereq: false@' \
     -i {helgrind,drd}/tests/pth_cond_destroy_busy.vgtest
```

Note

The OpenMP tests are skipped if libgomp has been compiled with `--enable-linux-futex` (the default). If needed, just recompile the libgomp library from the gcc build tree, passing `--disable-linux-futex` to `configure`, storing the library to some place and changing the link from `/usr/lib/libgomp.so.1` to point to the new library.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed -i ... docs/Makefile.in` : This sed provides for installing the documentation in a versioned directory.

`--enable-lto=yes`: This option allows building Valgrind with LTO (link time optimization). This produces a smaller/faster Valgrind (up to 10%), but build time increases to about 5.5 SBU.

Contents

Installed Programs: callgrind_annotate, callgrind_control, cg_annotate, cg_diff, cg_merge, ms_print, valgrind, valgrind-disk-server, valgrind-listener, and vgdb

Installed Libraries: None

Installed Directories: /usr/lib/valgrind, /usr/libexec/valgrind, /usr/include/valgrind, and /usr/share/doc/valgrind-3.23.0

Short Descriptions

<code>valgrind</code>	is a program for debugging and profiling Linux executables
<code>callgrind_annotate</code>	takes an output file produced by the Valgrind tool Callgrind and prints the information in an easy-to-read form
<code>callgrind_control</code>	controls programs being run by the Valgrind tool Callgrind
<code>cg_annotate</code>	is a post-processing tool for the Valgrind tool Cachegrind
<code>cg_diff</code>	compares two Cachegrind output files
<code>cg_merge</code>	merges multiple Cachegrind output files into one
<code>ms_print</code>	takes an output file produced by the Valgrind tool Massif and prints the information in an easy-to-read form
<code>valgrind-di-server</code>	is a server that reads debuginfo from objects stored on a different machine
<code>valgrind-listener</code>	listens on a socket for Valgrind commentary
<code>vgdb</code>	is an intermediary between Valgrind and GDB or a shell

yasm-1.3.0

Introduction to yasm

Yasm is a complete rewrite of the [NASM-2.16.03](#) assembler. It supports the x86 and AMD64 instruction sets, accepts NASM and GAS assembler syntaxes and outputs binary, ELF32 and ELF64 object formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.tortall.net/projects/yasm/releases/yasm-1.3.0.tar.gz>
- Download MD5 sum: fc9e586751ff789b34b1f21d572d96af
- Download size: 1.5 MB
- Estimated disk space required: 27 MB (additional 12 MB for the tests)
- Estimated build time: 0.1 SBU (additional 0.1 SBU for the tests)

yasm Dependencies

Optional

[cython-3.0.11](#) and [Python2](#)

Installation of yasm

Install yasm by running the following commands:

```
sed -i 's#) yasm.*#)##' Makefile.in &&
./configure --prefix=/usr &&
make
```

To test the results, issue: `make -j1 check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed -i 's#) yasm.*#)##' Makefile.in`: This sed prevents it compiling 2 programs (vsyasm and ytasm) that are only of use on Microsoft Windows.

Contents

Installed Program: yasm

Installed Library: libyasm.a

Installed Directory: /usr/include/libyasm

Short Descriptions

<code>yasm</code>	is a portable, retargetable assembler that supports the x86 and AMD64 instruction sets, accepts NASM and GAS assembler syntaxes and outputs binaries in ELF32 and ELF64 object formats
<code>libyasm.a</code>	provides all of the core functionality of <code>yasm</code> , for manipulating machine instructions and object file constructs

Java

Java-22.0.2

About Java

Java is different from most of the packages in LFS and BLFS. It is a programming language that works with files of bytecode to obtain instructions and executes them in a Java Virtual Machine (JVM). An introductory java program looks like:

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello, World");
    }
}
```

This program is saved as `HelloWorld.java`. The file name, `HelloWorld`, must match the class name. It is then converted into byte code with `javac HelloWorld.java`. The output file is `HelloWorld.class`. The program is executed with `java HelloWorld`. This creates a JVM and runs the code. The 'class' extension must not be specified.

Several class files can be combined into one file with the `jar` command. This is similar to the standard `tar` command. For instance, the command `jar cf myjar.jar *.class` will combine all class files in a directory into one jar file. These act as library files.

The JVM can search for and use classes in jar files automatically. It uses the `CLASSPATH` environment variable to search for jar files. This is a standard list of colon-separated directory names similar to the `PATH` environment variable.

Binary JDK Information

Creating a JVM from source requires a set of circular dependencies. The first thing that's needed is a set of programs called a Java Development Kit (JDK). This set of programs includes `java`, `javac`, `jar`, and several others. It also includes several base `jar` files.

To start, we set up a binary installation of the JDK created by the BLFS editors. It is installed in the `/opt` directory to allow for multiple installations, including a source based version.

This package is known to build and work properly using an LFS 12.2 platform.

Binary Package Information

- Binary download (x86): <https://anduin.linuxfromscratch.org/BLFS/OpenJDK/OpenJDK-22.0.2/OpenJDK-22.0.2+9-i686-bin.tar.xz>
- Download MD5 sum: 400033f7e0755c14687afe277a5f5672
- Download size (binary): 162 MB
- Estimated disk space required: 312 MB
- Binary download (x86_64): https://anduin.linuxfromscratch.org/BLFS/OpenJDK/OpenJDK-22.0.2/OpenJDK-22.0.2+9-x86_64-bin.tar.xz
- Download MD5 sum: 83f5b325922b3cef357a6e63d368ce91
- Download size (binary): 170 MB
- Estimated disk space required: 341 MB

Java Binary Runtime Dependencies

[alsa-lib-1.2.12](#), [Cups-2.4.10](#), [giflib-5.2.2](#), and [Xorg Libraries](#)

Installation of the Java BinaryJDK

Begin by extracting the appropriate binary tarball for your architecture and changing to the extracted directory. Install the binary OpenJDK with the following commands as the `root` user:

```
install -vdm755 /opt/OpenJDK-22.0.2-bin &&
mv -v * /opt/OpenJDK-22.0.2-bin &&
chown -R root:root /opt/OpenJDK-22.0.2-bin
```

The binary version is now installed. You may create a symlink to that version by issuing, as the `root` user:

```
ln -sfn OpenJDK-22.0.2-bin /opt/jdk
```

You may now proceed to [Configuring the Java environment](#), where the instructions assume that the above link exists.

OpenJDK-22.0.2

Introduction to OpenJDK

OpenJDK is an open-source implementation of Oracle's Java Standard Edition platform. OpenJDK is useful for developing Java programs, and provides a complete runtime environment to run Java programs.

This package is known to build and work properly using an LFS 12.2 platform.

OpenJDK is GPL'd code, with a special exception made for non-free projects to use these classes in their proprietary products. In similar fashion to the LGPL, which allows non-free programs to link to libraries provided by free software, the [GNU General Public License, version 2, with the Classpath Exception](#) allows third party programs to use classes provided by free software without the requirement that the third party software also be free. As with the LGPL, any modifications made to the free software portions of a third party application, must also be made freely available.

Note

The OpenJDK source includes a very thorough, open source test suite using the JTreg test harness. The testing instructions below allow testing the just built JDK for reasonable compatibility with the proprietary Oracle JDK. However, in order for an independent implementation to claim compatibility, it must pass a proprietary JCK/TCK test suite. No claims of compatibility, or even partial compatibility, may be made without passing an approved test suite.

Oracle does provide free community access, on a case-by-case basis, to a closed toolkit to ensure 100% compatibility with its proprietary JDK. Neither the binary version provided on the [Java-22.0.2](#) page nor the JVM built with the instructions below have been tested against the [TCK](#). Any version that is built using the instructions given, cannot claim to be compatible with the proprietary JDK, without the user applying for, and completing the compatibility tests themselves.

With that in mind, the binaries produced using this build method are regularly tested against the TCK by the members listed on the site above. In addition to the community license above, an educational, non-commercial license for the TCK can be obtained [here](#).

Package Information

- Download (HTTP): <https://github.com/openjdk/jdk22u/archive/jdk-22.0.2-ga.tar.gz>
- Download MD5 sum: 13b4e71252055e6d78ae12b8db5021d4
- Download Size: 107 MB
- Estimated disk space required: 3.7 GB (add 625 MB for tests)
- Estimated build time: 5.2 SBU with 4 jobs (add 34 SBU for tests with 4 jobs)

Additional Downloads

Optional test harness

- <https://anduin.linuxfromscratch.org/BLFS/OpenJDK/OpenJDK-22.0.2/jtreg-7.3.1+1.tar.gz>
- Download MD5 sum: 0038551ecaf37d0cd99832217f79e56d
- Download Size: 9.0 MB

OpenJDK Dependencies

Required Dependencies

An existing binary ([Java-22.0.2](#) or an earlier built version of this package. The instructions below assume that you are using [Configuring the Java environment](#)), [alsa-lib-1.2.12](#), [cpio-2.15](#), [Cups-2.4.10](#), [UnZip-6.0](#), [Which-2.21](#), [Xorg Libraries](#), and [Zip-3.0](#)

Recommended

[make-ca-1.14](#), [giflib-5.2.2](#), [harfBuzz-9.0.0](#), [Little CMS-2.16](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), and [Wget-1.24.5](#)

Optional

[git-2.46.0](#), [Graphviz-12.1.0](#), [Mercurial-6.8.1](#), [ccache](#), [pandoc](#), and [pigz](#)

Installation of OpenJDK

If you have downloaded the optional test harness, unpack it now:

```
tar -xf ./jtreg-7.3.1+1.tar.gz
```

Note

Before proceeding, you should ensure that your environment `PATH` variable contains the location of the Java compiler used for bootstrapping OpenJDK. This is the only requirement for the environment. Modern Java installations do not need `JAVA_HOME` and `CLASSPATH` is not used here. Furthermore, OpenJDK developers recommend unsetting `JAVA_HOME`.

The build system does not allow the `-j` switch in `MAKEFLAGS`. See the command explanation for `--with-jobs=` for more information on customizing parallelization.

Configure and build the package with the following commands:

```
unset JAVA_HOME          &&
bash configure --enable-unlimited-crypto \
    --disable-warnings-as-errors \
    --with-stdc++-lib=dynamic \
    --with-giflib=system \
    --with-harfbuzz=system \
    --with-jtreg=$PWD/jtreg \
    --with-lcms=system \
    --with-libjpeg=system \
    --with-libpng=system \
    --with-zlib=system \
    --with-version-build="9" \
    --with-version-pre="" \
    --with-version-opt="" \
    --with-cacerts-file=/etc/pki/tls/java/cacerts &&
make images
```

To test the results, you will need to execute the `jtreg` program. You can set the number of concurrent tests by adding the `-conc:<X>` value in the below command (tests will run sequentially otherwise):

```
export JT_JAVA=$(echo $PWD/build/*/jdk) &&
jtreg/bin/jtreg -jdk:$JT_JAVA -automatic -ignore:quiet -v1 \
    test/jdk:tier1 test/langtools:tier1 &&
unset JT_JAVA
```

For more control over the test suite, review the documentation available in [jtreg/doc/jtreg/usage.txt](#). To review the results, see the files `JTreport/test_{jdk,langtools}/text/stats.txt` and `JTreport/test_{jdk,langtools}/text/summary.txt`. You should expect to see around 60 failures and 10 errors.

Install the package with the following commands as the `root` user:

```
install -vdm755 /opt/jdk-22.0.2+9          &&
cp -Rv build/*/*images/jdk/* /opt/jdk-22.0.2+9 &&
chown -R root:root /opt/jdk-22.0.2+9          &&
for s in 16 24 32 48; do
    install -vDm644 src/java/desktop/unix/classes/sun.awt/X11/java-icon${s}.png \
        /usr/share/icons/hicolor/${s}x${s}/apps/java.png
done
```

Note

If you only wish to install the Java Runtime Environment, you can substitute `build/*/images/jre` in the above `cp` command.

There are now two OpenJDK SDKs installed in `/opt`. You should decide on which one you would like to use as the default. Normally, you would opt for the just installed OpenJDK. If so, do the following as the `root` user:

```
ln -v -nsf jdk-22.0.2+9 /opt/jdk
```

If desired, you can create `.desktop` files to add entries in the menu for `java` and `jconsole`. The needed icons have already been installed. As the `root` user:

```
mkdir -pv /usr/share/applications &&  
  
cat > /usr/share/applications/openjdk-java.desktop << "EOF" &&  
[Desktop Entry]  
Name=OpenJDK Java 22.0.2 Runtime  
Comment=OpenJDK Java 22.0.2 Runtime  
Exec=/opt/jdk/bin/java -jar  
Terminal=false  
Type=Application  
Icon=java  
MimeType=application/x-java-archive;application/java-archive;application/x-jar;  
NoDisplay=true  
EOF  
cat > /usr/share/applications/openjdk-jconsole.desktop << "EOF"  
[Desktop Entry]  
Name=OpenJDK Java 22.0.2 Console  
Comment=OpenJDK Java 22.0.2 Console  
Keywords=java;console;monitoring  
Exec=/opt/jdk/bin/jconsole  
Terminal=false  
Type=Application  
Icon=java  
Categories=Application;System;  
EOF
```

Command Explanations

`bash configure...`: the top level `configure` is a wrapper around the autotools one. It is not executable and must be run through `bash`.

`--enable-unlimited-crypto`: Because of limitations on the usage of cryptography in some countries, there is the possibility to limit the size of encryption keys and the use of some algorithms in a policy file. This switch allows to ship a policy file with no restriction. It is the responsibility of the user to ensure proper adherence to the law.

`--disable-warnings-as-errors`: This switch disables use of `-Werror` in the build.

`--with-stdc++lib=dynamic`: This switch forces the build system to link to `libstdc++.so` (dynamic) instead of `libstdc++.a` (static).

`--with-jobs=<X>`: The `-j` passed to make does not work with make as invoked here. By default, the build system will use the number of CPUs - 1.

`--with-jtreg=$PWD/jtreg`: This switch tells configure where to find jtreg. Omit if you have not downloaded the optional test suite.

`--with-(giflib,harfbuzz,lcms,libjpeg,libpng,zlib)=system`: These switches force the build system to use the system libraries instead of the bundled versions.

`--with-version-build`: Currently, the build system does not include the build number in the version string. It has to be specified here.

`--with-version-pre`: This switch allows you to prefix the version string with a custom string.

`--with-version-opt`: This switch allows you to add an optional build description to the version string.

`--with-cacerts-file=/etc/pki/tls/java/cacerts`: Specifies where to find a `cacerts` file, `/etc/pki/tls/java/` on a BLFS system. Otherwise, an empty one is created. You can use the `/usr/sbin/make-ca --force` command to generate it, once you have installed the Java binaries.

--with-boot-jdk: This switch provides the location of the temporary JDK. It is normally not needed if `java` is found in the `PATH`.

Configuring OpenJDK

Configuration Information

Normally, the Java environment has been configured after installing the binary version, and can be used with the just built package as well. Review [Configuring the Java environment](#) in case you want to modify something.

To test if the man pages are correctly installed, issue `source /etc/profile` and `man java` to display the respective man page.

Setting up the JRE Certificate Authority Certificates (cacerts) file

If you have run the instructions on the [make-ca-1.14](#) page, you only need to create a symlink in the default location for the `cacerts` file. As user `root`:

```
ln -s /etc/pki/tls/java/cacerts /opt/jdk/lib/security/cacerts
```

To check the installation, issue:

```
cd /opt/jdk  
bin/keytool -list -cacerts
```

At the prompt `Enter keystore password:`, enter `changeit` (the default) or just press the "Enter" key. If the `cacerts` file was installed correctly, you will see a list of the certificates with related information for each one. If not, you need to reinstall them.

Contents

- Installed Programs:** jar, jarsigner, java, javac, javadoc, javap, jcmd, jconsole, jdb, jdeprscan, jdeps, jfr, jhsdb, jimage, jinfo, jlink, jmap, jmod, jpackage, jps, jrunscript, jshell, jstack, jstat, jstatd, jwebserver, keytool, rmiregistry, and serialver
- Installed Libraries:** 38 libraries in /opt/jdk-22.0.2+9/lib/
- Installed Directory:** /opt/jdk-22.0.2+9

Short Descriptions

<code>jar</code>	combines multiple files into a single jar archive
<code>jarsigner</code>	signs jar files and verifies the signatures and integrity of a signed jar file
<code>java</code>	launches a Java application by starting a Java runtime environment, loading a specified class and invoking its main method
<code>javac</code>	reads class and interface definitions, written in the Java programming language, and compiles them into bytecode class files
<code>javadoc</code>	parses the declarations and documentation comments in a set of Java source files and produces a corresponding set of HTML pages describing the classes, interfaces, constructors, methods, and fields
<code>javap</code>	disassembles a Java class file
<code>jcmd</code>	is a utility to send diagnostic command requests to a running Java Virtual Machine
<code>jconsole</code>	is a graphical console tool to monitor and manage both local and remote Java applications and virtual machines
<code>jdb</code>	is a simple command-line debugger for Java classes
<code>jdeprscan</code>	scans class or jar files for uses of deprecated API elements
<code>jdeps</code>	shows the package-level or class-level dependencies of Java class files
<code>jfr</code>	is a tool for working with "Flight Recorder" files
<code>jhsdb</code>	is a tool to analyze the content of a core dump from a crashed Java Virtual Machine (JVM)
<code>jimage</code>	is used to list, extract, verify, or get information about modules in <code>jimage</code> format
<code>jinfo</code>	prints Java configuration information for a given Java process, core file, or a remote debug server
<code>jlink</code>	is used to assemble and optimize a set of modules and their dependencies into a custom runtime image
<code>jmap</code>	prints shared object memory maps or heap memory details of a given process, core file, or a remote debug server
<code>jmod</code>	creates JMOD files and lists the content of existing JMOD files
<code>jpackage</code>	generates java application packages and images

<code>jps</code>	lists the instrumented JVMs on the target system
<code>jrunscript</code>	is a command line script shell
<code>jshell</code>	is an interactive tool for learning the Java programming language and prototyping Java code
<code>jstack</code>	prints Java stack traces of Java threads for a given Java process, core file, or a remote debug server
<code>jstat</code>	displays performance statistics for an instrumented JVM
<code>jstard</code>	is an RMI server application that monitors for the creation and termination of instrumented JVMs
<code>jwebserver</code>	provides a minimal HTTP server, designed to be used for prototyping, testing, and debugging
<code>keytool</code>	is a key and certificate management utility
<code>rmiregistry</code>	creates and starts a remote object registry on the specified port on the current host
<code>serialver</code>	returns the serialVersionUID for one or more classes in a form suitable for copying into an evolving class

Configuring the Java environment

Setting up the environment

After the package installation is complete, the next step is to make sure that the system can properly find the files. If you set up your login scripts as recommended in [The Bash Shell Startup Files](#), update the environment by creating the `openjdk.sh` script, as the `root` user:

```
cat > /etc/profile.d/openjdk.sh << "EOF"
# Begin /etc/profile.d/openjdk.sh

# Set JAVA_HOME directory
JAVA_HOME=/opt/jdk

# Adjust PATH
pathappend $JAVA_HOME/bin

# Auto Java CLASSPATH: Copy jar files to, or create symlinks in, the
# /usr/share/java directory.

AUTO_CLASSPATH_DIR=/usr/share/java

pathprepend . CLASSPATH

for dir in `find ${AUTO_CLASSPATH_DIR} -type d 2>/dev/null`; do
    pathappend $dir CLASSPATH
done

for jar in `find ${AUTO_CLASSPATH_DIR} -name "*.jar" 2>/dev/null`; do
    pathappend $jar CLASSPATH
done

export JAVA_HOME

# By default, Java creates several files in a directory named
# /tmp/hspersfdata_[username]. This directory contains files that are used for
# performance monitoring and profiling, but aren't normally needed on a BLFS
# system. This environment variable disables that feature.
_JAVA_OPTIONS="-XX:-UsePerfData"

export _JAVA_OPTIONS

unset AUTO_CLASSPATH_DIR dir jar _JAVA_OPTIONS

# End /etc/profile.d/openjdk.sh
EOF
```

If [Sudo-1.9.15p5](#) is installed, the super user should have access to the above variables. Execute the following commands as the `root` user:

```
cat > /etc/sudoers.d/java << "EOF"
Defaults env_keep += JAVA_HOME
Defaults env_keep += CLASSPATH
Defaults env_keep += _JAVA_OPTIONS
EOF
```

To use `mandb` to include the OpenJDK man pages in its database, issue, as the `root` user:

```
cat >> /etc/man_db.conf << "EOF" &&
# Begin Java addition
MANDATORY_MANPATH      /opt/jdk/man
MANPATH_MAP             /opt/jdk/bin      /opt/jdk/man
MANDB_MAP               /opt/jdk/man     /var/cache/man/jdk
# End Java addition
EOF

mkdir -p /var/cache/man &&
mandb -c /opt/jdk/man
```

Setting up the Certificate Authority Certificates for Java

OpenJDK uses its own format for the CA certificates. The Java security modules use `$JAVA_HOME/lib/security/cacerts` by default. In order to keep all the certificates in one place, we use `/etc/ssl/java/cacerts`. The instructions on the [make-ca-1.14](#) page previously created the file located in `/etc/ssl/java`. Set up a symlink in the default location as the `root` user:

```
ln -sfv /etc/pki/tls/java/cacerts /opt/jdk/lib/security/cacerts
```

Use the following command to check if the `cacerts` file has been successfully installed:

```
/opt/jdk/bin/keytool -list -cacerts
```

At the prompt `Enter keystore password:`, enter `changeit` (the default) or just press the “Enter” key. If the `cacerts` file was installed correctly, you will see a list of the certificates with related information for each one. If not, you need to reinstall them.

If you later install a new JVM, you just have to create the symlink in the default location to be able to use the `cacerts`.

apache-ant-1.10.14

Introduction to Apache Ant

The Apache Ant package is a Java-based build tool. In theory, it is like the `make` command, but without `make`'s wrinkles. Ant is different. Instead of a model that is extended with shell-based commands, Ant is extended using Java classes. Instead of writing shell commands, the configuration files are XML-based, calling out a target tree that executes various tasks. Each task is run by an object that implements a particular task interface.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist/ant/source/apache-ant-1.10.14-src.tar.xz>
- Download MD5 sum: 055843219f487edb3a6db554ad1355ef
- Download size: 3.6 MB
- Estimated disk space required: 195 MB
- Estimated build time: 0.3 SBU (excluding download time)

Apache Ant Dependencies

Required

A JDK ([Java Binary](#) or [OpenJDK-22.0.2](#)) and [GLib-2.80.4](#)

Note

An Internet connection is needed for building this package.

Installation of Apache Ant

Build a limited bootstrap version of Apache Ant using the following command:

```
./bootstrap.sh
```

Download the runtime dependencies using the `fetch.xml` ant build script:

```
bootstrap/bin/ant -f fetch.xml -Ddest=optional
```

Build Apache Ant by running the following command:

```
./build.sh -Ddist.dir=$PWD/ant-1.10.14 dist
```

Install, as the `root` user:

```
cp -rv ant-1.10.14 /opt/          &&
chown -R root:root /opt/ant-1.10.14 &&
ln -sfv ant-1.10.14 /opt/ant
```

Command Explanations

`bootstrap/bin/ant -f fetch.xml -D dest=optional`: Downloads the missing dependencies to the user's home directory, and copies them into the source tree (in the `lib/optional` directory, where `ant` picks them up at build time).

`./build.sh -D dist.dir=$PWD/ant-1.10.14 dist`: This command builds, tests, then installs the package into a temporary directory.

Configuring Apache Ant

Config Files

`/etc/ant/ant.conf`, `~/.ant/ant.conf`, and `~/.antrc`

Configuration Information

Some packages will require `ant` to be in the search path and the `ANT_HOME` environment variable defined. Satisfy these requirements by issuing, as the `root` user:

```
cat > /etc/profile.d/ant.sh << EOF
# Begin /etc/profile.d/ant.sh

pathappend /opt/ant/bin
export ANT_HOME=/opt/ant

# End /etc/profile.d/ant.sh
EOF
```

The above instructions assume you have configured your system as described in [The Bash Shell Startup Files](#).

Contents

Installed Programs: `ant`, `antRun`, `antRun.pl`, `complete-ant-cmd.pl`, `runant.pl`, and `runant.py`

Installed Libraries: Numerous `ant*.jar` and dependent libraries in `$ANT_HOME/lib`

Installed Directories: `/opt/ant-1.10.14`

Short Descriptions

<code>ant</code>	is a Java based build tool used by many packages instead of the conventional <code>make</code> program
<code>antRun</code>	is a support script used to start <code>ant</code> build scripts in a given directory
<code>antRun.pl</code>	is a Perl script that provides similar functionality offered by the <code>antRun</code> script
<code>complete-ant-cmd.pl</code>	is a Perl script that allows Bash to complete an <code>ant</code> command-line
<code>runant.pl</code>	is a Perl wrapper script used to invoke <code>ant</code>
<code>runant.py</code>	is a Python wrapper script used to invoke <code>ant</code>
<code>ant*.jar</code>	files are the Apache Ant Java class libraries

Part IV. Networking

Chapter 14. Connecting to a Network

The LFS book covers setting up networking by connecting to a LAN with a static IP address. There are other methods used to obtain an IP address and connect to a LAN and other networks (such as the Internet). The most popular methods (DHCP and PPP) are covered in this chapter.

DHCP stands for Dynamic Host Configuration Protocol. It is a protocol used by many sites to automatically provide information such as IP addresses, subnet masks and routing information to computers. If your network uses DHCP, you will need a DHCP client in order to connect to it.

Advanced Network Setup

Network Bridge

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
Networking support: Y
Networking options:
 802.1d Ethernet Bridging: M or Y
```

Setting up a Network Bridge

In this section we are going to discuss how to set up a network bridge using `systemd-networkd`. In the examples below, `eth0` represents the external interface that is being bridged, while `br0` represents the bridge interface.

To create a bridge interface, create the following configuration file by running the following command as the `root` user:

```
cat > /etc/systemd/network/50-br0.netdev << EOF
[NetDev]
Name=br0
Kind=bridge
EOF
```

To assign a network interface to a bridge, create the following configuration file by running the following command as the `root` user:

```
cat > /etc/systemd/network/51-eth0.network << EOF
[Match]
Name=eth0

[Network]
Bridge=br0
EOF
```

Repeat the process for any other interfaces that need to be bridged. Note that it is important that nothing assigns any addresses to the bridged interfaces. If you are using [NetworkManager-1.48.8](#), make sure you configure them to ignore the bridged interfaces, as well as the bridge interface itself.

If you are on a network which uses DHCP for assigning ip addresses, create the following configuration file by running the following command as the `root` user:

```
cat > /etc/systemd/network/60-br0.network << EOF
[Match]
Name=br0

[Network]
DHCP=yes
EOF
```

Alternatively, if using a static ip setup, create the following configuration file by running the following command as the `root` user:

```
cat > /etc/systemd/network/60-br0.network << EOF
[Match]
Name=br0

[Network]
```

```
Address=192.168.0.2/24  
Gateway=192.168.0.1  
DNS=192.168.0.1  
EOF
```

To bring up the bridge interface, simply restart the `systemd-networkd` daemon by running the following command as the `root` user:

```
sudo systemctl restart systemd-networkd
```

dhcpcd-10.0.8

Introduction to dhcpcd

`dhcpcd` is an implementation of the DHCP client specified in RFC2131. A DHCP client is useful for connecting your computer to a network which uses DHCP to assign network addresses. `dhcpcd` strives to be a fully featured, yet very lightweight DHCP client.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/NetworkConfiguration/dhcpcd/releases/download/v10.0.8/dhcpcd-10.0.8.tar.xz>
- Download MD5 sum: 8b813685a48ab017bcfb5e720a9f0181
- Download size: 265.7 KB
- Estimated disk space required: 3.0 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

dhcpcd Dependencies

Optional

[LLVM-18.1.7](#) (with Clang), [ntp-4.2.8p18](#), [chrony](#), and [ypbind](#)

Privilege separation

Recent releases of `dhcpcd` optionally support privilege separation. As the practical security benefits of this are unclear for a program like `dhcpcd` and the setup is more complicated, the book currently defaults to disable it.

If you however would like to use privilege separation, additional installation steps are necessary to set up the proper environment. Issue the following commands as the `root` user:

```
install -v -m700 -d /var/lib/dhcpcd &&  
  
groupadd -g 52 dhcpcd &&  
useradd -c 'dhcpcd PrivSep' \  
-d /var/lib/dhcpcd \  
-g dhcpcd \  
-s /bin/false \  
-u 52 dhcpcd &&  
chown -v dhcpcd:dhcpcd /var/lib/dhcpcd
```

Installation of dhcpcd

Build `dhcpcd` without privilege separation by running the following command:

```
./configure --prefix=/usr \  
--sysconfdir=/etc \  
--libexecdir=/usr/lib/dhcpcd \  
--dbdir=/var/lib/dhcpcd \  
--runstatedir=/run \  
--disable-privsep &&  
make
```

Alternatively, build `dhcpcd` with privilege separation by running the following commands:

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --libexecdir=/usr/lib/dhcpd \
            --dbdir=/var/lib/dhcpd \
            --runstatedir=/run \
            --privsepuser=dhcpd &&
make
```

To test the results, issue: `make test`.

Now, as the *root* user:

make install

Command Explanations

--libexecdir=/usr/lib/dhcpcd: This switch sets a better location for the dhcpcd internal libraries.

`--dbdir=/var/lib/dhcpd`: This switch adjusts the database directory because the default directory, `/var/db`, is not FHS-compliant.

`--runstatedir=/run`: This switch sets the runtime state directory because the default `/var/run` is a symbolic link to `/run`, and using `/var/run` is deprecated.

--disable-privsep: This switch disables privilege separation, which is the default in dhpcd. This switch is not used in the build configuration where privilege separation is used.

`--privsepuser=dhcpca`: This switch sets the privilege separation user in the build configuration where privilege escalation is used.

--with-hook=...: You can optionally install more hooks, for example to install some configuration files such as `ntp.conf`. A set of hooks can be found in the `dhcpcd-hooks` directory in the build tree.

Configuring dhcpcd

Config Files

/etc/dhcpcd.conf

General Configuration Information

If you want to configure network interfaces at boot using `dhcpcd`, you need to install the `systemd` unit included in `blfs-systemd-units-20240801` package by running the following command as the `root` user:

```
make install-dhcpcd
```

Note

The default behavior of `dhcpd` is to set the hostname and the mtu. It also overwrites `/etc/resolv.conf` and `/etc/ntp.conf`. These modifications to system configuration files are done by hooks which are stored in `/usr/lib/dhcpd/dhcpd-hooks`. You can change this behavior by removing or adding hooks from/to that directory. The execution of hooks can be disabled by using the `--nohook (-c)` command line option or by the `nohook` option in the `/etc/dhcpd.conf` file.

Note

Make sure that you disable the `systemd-networkd` service or configure it not to manage the interfaces you want to manage with `dhcpcd`.

At this point you can test if `dhcpcd` is behaving as expected by running the following command as the `root` user:

```
systemctl start dhcpcd@eth0
```

To start `dmcrypt` on a specific interface at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable dhcpcd@eth0
```

Replace `eth0` with the actual interface name.

Contents

Installed Program: dhcpcd

Installed Library: /usr/lib/dhcpcd/dev/udev.so

Installed Directory: /{usr,var}/lib/dhcpcd and /usr/share/dhcpcd

Short Descriptions

`dhcpcd` is an implementation of the DHCP client specified in RFC2131

`udev.so` adds udev support for interface arrival and departure; this is because udev likes to rename the interface, which it can't do if dhcpcd grabs it first

Chapter 15. Networking Programs

These applications are generally client applications used to access the appropriate server across the building or across the world. Tcpwrappers and portmap are support programs for daemons that you may have running on your machine.

bridge-utils-1.7.1

Introduction to bridge-utils

The bridge-utils package contains a utility needed to create and manage bridge devices. This is useful in setting up networks for a hosted virtual machine (VM).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/utils/net/bridge-utils/bridge-utils-1.7.1.tar.xz>
- Download MD5 sum: 3e1fee4dc22cac5457c2f6ff990a518
- Download size: 29 KB
- Estimated disk space required: 1.1 MB
- Estimated build time: less than 0.1 SBU

bridge-utils Dependencies

Optional (to run tests)

[Net-tools-2.10](#)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
[*] Networking support ---> [NET]
    Networking options --->
        <*/M> 802.1d Ethernet Bridging [BRIDGE]
```

Installation of bridge-utils

Install bridge-utils by running the following commands:

```
autoconf      &&
./configure --prefix=/usr &&
make
```

Testing the results requires running the six shell scripts in the `tools/` directory. Two of the tests require two ethernet ports. Some tests will not preserve the current network configuration. See `tests/README` for details.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: brctl

Installed Libraries: None

Installed Directories: None

Short Descriptions

`brctl` is a program used to set up, maintain, and inspect the ethernet bridge configuration in the linux kernel

cifs-utils-7.0

Introduction to cifs-utils

The cifs-utils package provides a means for mounting SMB/CIFS shares on a Linux system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.samba.org/ftp/linux-cifs/cifs-utils/cifs-utils-7.0.tar.bz2>
- Download MD5 sum: 518431bf43f23e6aacd97e80e2060df7
- Download size: 412 KB
- Estimated disk space required: 4.2 MB
- Estimated build time: 0.1 SBU

cifs-utils Dependencies

Recommended

[MIT Kerberos V5-1.21.3](#) and [Talloc-2.4.2](#) (required if [MIT Kerberos V5-1.21.3](#) is installed)

Optional

[docutils-0.21.2](#) (to create the man pages), [keyutils-1.6.3](#) (required to build PAM module), [Linux-PAM-1.6.1](#), [Samba-4.20.4](#), and [libcap-2.70 with PAM](#) or [libcap-ng](#)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
File systems --->
 [*] Network File Systems --->                                [NETWORK_FILESYSTEMS]
   <*/M> SMB3 and CIFS support (advanced network filesystem)      [CIFS]
```

Depending on your server configuration, additional kernel options may be required.

Installation of cifs-utils

Install cifs-utils by running the following commands:

```
./configure --prefix=/usr \
            --disable-pam &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-pam`: Do not build PAM support. Remove it and use `--with-pamdir` (see below), if [Linux-PAM-1.6.1](#) is installed and you wish PAM support.

`--with-pamdir=/usr/lib/security`: Install the PAM module in `/usr/lib/security`.

Contents

Installed Programs: `cifs.idmap`, `cifs.upcall`, `cifscreds`, `getcifsacl`, `mount.cifs`, `mount.smb3`, `setcifsacl`, `smb2-quota`, and `smbinfo`

Installed Library: `/usr/lib/cifs-utils/idmapwb.so` and optionally PAM module `/usr/lib/security/pam_cifscreds.so`

Installed Directory: `/usr/lib/cifs-utils`

Short Descriptions

<code>cifs.idmap</code>	is a userspace helper program for the linux CIFS client filesystem. There are a number of activities that the kernel cannot easily do itself. This program is a callout program that does these things for the kernel and then returns the result. It is not intended to be run from the command-line
<code>cifs.upcall</code>	is a userspace helper program for the linux CIFS client filesystem. It is intended to be run when the kernel calls request-key for a particular key type. It is not intended to be run from the command-line
<code>cifscreds</code>	is a tool for managing credentials (username and password) for the purpose of establishing sessions in multiuser mounts
<code>getcifsacl</code>	is a userspace helper to display an ACL in a security descriptor for Common Internet File System (CIFS)
<code>mount.cifs</code>	mounts a Linux CIFS filesystem. It is usually invoked indirectly by the <code>mount(8)</code> command when using the <code>-t cifs</code> option
<code>mount.smb3</code>	mounts a SMB3-based filesystem. It is usually invoked indirectly by the <code>mount(8)</code> command when using the <code>-t smb3</code> option
<code>setcifsacl</code>	is intended to alter an ACL of a security descriptor for a file system object
<code>smb3-quota</code>	displays quota information for a SMB filesystem
<code>smbinfo</code>	displays SMB-specific file information, such as security descriptors and quotas

NcFTP-3.2.7

Introduction to NcFTP

The NcFTP package contains a powerful and flexible interface to the Internet standard File Transfer Protocol. It is intended to replace or supplement the stock `ftp` program.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.ncftp.com/downloads/ncftp/ncftp-3.2.7-src.tar.xz>
- Download MD5 sum: `bcbc048d2412f4d62bc798818e703680`
- Download size: 416 KB
- Estimated disk space required: 6.9 MB
- Estimated build time: 0.2 SBU

Installation of NcFTP

There are two ways to build NcFTP. The first (and optimal) way builds most of the functionality as a shared library and then builds and installs the program linked against this library. The second method simply links all of the functionality into the binary statically. This doesn't make the dynamic library available for linking by other applications. You need to choose which method best suits you. Note that the second method does *not* create an entirely statically linked binary; only the `libncftp` parts are statically linked in, in this case. Be aware that building and using the shared library is covered by the Clarified Artistic License; however, developing applications that utilize the shared library is subject to a different license.

First, fix an issue with the configure script introduced by gcc-14:

```
sed -i 's/def HAVE_STDLIB_H/ 1;/s/extern select/extern int select/' configure
```

To install NcFTP using the first (and optimal) method, run the following commands:

```
CC=/usr/bin/gcc \
./configure --prefix=/usr --sysconfdir=/etc &&
make -C libncftp shared &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make -C libncftp soinstall &&
make install
```

To install NcFTP using the second method (with the `libncftp` functionality linked in statically) run the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`CC=/usr/bin/gcc`. This environment variable ensures that `gcc` is used if [LLVM-18.1.7](#) is installed. The build procedure is broken if `gcc` is not used.

`make -C ... && make -c ...`: These commands make and install the dynamic library `libncftp` which is then used to link against when compiling the main program.

Configuring NcFTP

Config Files

`/etc/ncftp.*` and `~/.ncftp/*`; especially `/etc/ncftp.prefs_v3` and `~/.ncftp/prefs_v3`

Configuration Information

Most NcFTP configuration is done while in the program, and the configuration files are dealt with automatically. One exception to this is `~/.ncftp/prefs_v3`. There are various options to alter in there, including:

```
yes-i-know-about-NcFTPD=yes
```

This disables the splash screen advertising the NcFTPD server.

There are other options in the `prefs_v3` file. Most of these are self-explanatory. Global defaults can be set in `/etc/ncftp.prefs_v3`.

Contents

Installed Programs: ncftp, ncftpbatch, ncftpbookmarks, ncftpget, ncftpls, ncftpput, and ncftpspooler

Installed Library: libncftp.so

Installed Directories: None

Short Descriptions

<code>ncftp</code>	is a browser program for File Transfer Protocol
<code>ncftpbatch</code>	is an individual batch FTP job processor
<code>ncftpbookmarks</code>	is the NcFTP Bookmark Editor (NCurses -based)
<code>ncftpget</code>	is an internet file transfer program for scripts used to retrieve files
<code>ncftpls</code>	is an internet file transfer program for scripts used to list files
<code>ncftpput</code>	is an internet file transfer program for scripts used to transfer files

Net-tools-2.10

Introduction to Net-tools

The Net-tools package is a collection of programs for controlling the network subsystem of the Linux kernel.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/project/net-tools/net-tools-2.10.tar.xz>
- Download MD5 sum: 78aae762c95e2d731faf88d482e4cde5
- Download size: 228 KB
- Estimated disk space required: 7.5 MB
- Estimated build time: less than 0.1 SBU

Installation of Net-tools

The instructions below automate the configuration process by piping `yes` to the `make` command. If you wish to run the interactive configuration process (by changing the instruction to just `make`), but you are not sure how to answer all the questions, then just accept the defaults. This will be just fine in the majority of cases. What you're asked here is a bunch of questions about which network protocols you've enabled in your kernel. The default answers will enable the tools from this package to work with the most common protocols: TCP, PPP, and several others. You still need to actually enable these protocols in the kernel—what you do here is merely tell the package to include support for those protocols in its programs, but it's up to the kernel to make the protocols available.

Note

This package has several unneeded protocols and hardware device specific functions that are obsolete. To only build the minimum needed for your system, skip the `yes` command and answer each question interactively. The minimum needed options are 'UNIX protocol family' and 'INET (TCP/IP) protocol family'.

For this package, we use the DESTDIR method of installation in order to easily remove files from the build that overwrite those that we want to keep or are not appropriate for our system.

Install Net-tools by running the following commands:

```
export BINDIR='/usr/bin' SBINDIR='/usr/bin' &&
yes "" | make -j1 &&
make DESTDIR=$PWD/install -j1 install &&
rm -install/usr/bin/{nis,yp}domainname &&
rm -install/usr/bin/{hostname,dnsdomainname,domainname,ifconfig} &&
rm -r install/usr/share/man/man1 &&
rm -install/usr/share/man/man8/ifconfig.8 &&
unset BINDIR SBINDIR
```

This package does not come with a test suite.

Now, as the `root` user:

```
chown -R root:root install &&
cp -a install/* /
```

Command Explanations

`export BINDIR='/usr/bin' SBINDIR='/usr/bin'`: Ensure the executables are installed in the correct location.

`yes "" | make` : Piping `yes` to `make config` skips the interactive configuration and accepts the defaults.

`rm ...`: Remove unneeded programs and man pages.

Contents

Installed Programs: arp, ipmaddr, iptunnel, mii-tool, nameif, netstat, plipconfig, rarp, route, and slattach

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>arp</code>	is used to manipulate the kernel's ARP cache, usually to add or delete an entry, or to dump the entire cache
<code>ipmaddr</code>	adds, deletes and shows an interface's multicast addresses
<code>iptunnel</code>	adds, changes, deletes and shows an interface's tunnels
<code>mii-tool</code>	checks or sets the status of a network interface's Media Independent Interface (MII) unit
<code>nameif</code>	names network interfaces based on MAC addresses
<code>netstat</code>	is used to report network connections, routing tables, and interface statistics
<code>plipconfig</code>	is used to fine tune the PLIP device parameters, to improve its performance
<code>rarp</code>	is used to manipulate the kernel's RARP table
<code>route</code>	is used to manipulate the IP routing table
<code>slattach</code>	attaches a network interface to a serial line. This allows you to use normal terminal lines for point-to-point links to others computers

NFS-Utils-2.6.4

Introduction to NFS Utilities

The NFS Utilities package contains the userspace server and client tools necessary to use the kernel's NFS abilities. NFS is a protocol that allows sharing file systems over the network.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/utils/nfs-utils/2.6.4/nfs-utils-2.6.4.tar.xz>
- Download MD5 sum: 907f95977ccf7a522ee32af1534f0e4c
- Download size: 712 KB
- Estimated disk space required: 18 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

NFS Utilities Dependencies

Required

[libtirpc-1.3.5](#), [libevent-2.1.12](#), [rpcsvc-proto-1.4.4](#), and [SQLite-3.46.1](#)

Optional

[Cyrus SASL-2.1.28](#) (for SASL authentication), [LVM2-2.03.26](#) (libdevmapper for NFSv4 support), [libnsl-2.0.1](#) (for NIS client support), [OpenLDAP-2.6.8](#) (for LDAP authentication), [MIT Kerberos V5-1.21.3](#) or [libgssapi](#), and [librpcsecgss](#) (for GSS and RPC security support), and [libcap-2.70 with PAM](#)

Required (runtime)

[rpcbind-1.2.7](#)

Kernel Configuration

Enable the following options in the kernel configuration (choose client and/or server support as appropriate) and recompile the kernel if necessary:

```
File systems --->
[*] Network File Systems --->                                [NETWORK_FILESYSTEMS]
  <*/M> NFS client support                                     [NFS_FS]
  <*/M> NFS server support                                    [NFSD]
```

Select the appropriate sub-options that appear when the above options are selected.

Note

In BLFS we assume that nfs v3 will be used. If the *server* offers nfs v4 (for linux, CONFIG_NFSD_V4) then auto-negotiation for v3 will fail and you will need to add `nfsver=3` to the mount options. This also applies if that option is enabled in the *client's* kernel, for example in a distro trying to mount from a BLFS v3 server.

Even if neither end of the connection supports nfs v4, adding `nfsver=3` is still beneficial because it prevents an error message "NFS: bad mount option value specified: minorversion=1" being logged on every mount.

Installation of NFS Utilities

Install NFS Utilities by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --sbindir=/usr/sbin \
            --disable-nfsv4   \
            --disable-gss     \
LIBS="-lsqlite3 -levent_core" &&
make
```

Now, as the `root` user:

```
make install      &&
chmod u+w,go+r /usr/sbin/mount.nfs &&
chown nobody:nogroup /var/lib/nfs
```

The tests for this package require that the package be installed. In addition, the rpc.statd daemon must not be running and the tests need to be run as the `root` user.

To test the results, issue, as `root`:

```
make check
```

Command Explanations

`--disable-gss`: Disables support for RPCSEC GSS (RPC Security).

`LIBS="-lsqlite3 -levent_core"`: is required for the fsidd program.

`chown nobody:nogroup /var/lib/nfs`: The rpc.statd program uses the ownership of this directory to set its UID and GID. This command sets those to unprivileged entries.

Configuring NFS Utilities

Server Configuration

`/etc/exports` contains the exported directories on NFS servers. Refer to the `exports.5` manual page for the syntax of this file. Also refer to the "NFS HowTo" available at <https://nfs.sourceforge.net/nfs-howto/> for information on how to configure the servers and clients in a secure manner. For example, for sharing the `/home` directory over the local network, the following line may be added:

```
cat >> /etc/exports << EOF
/home 192.168.0.0/24(rw,subtree_check,anonuid=99,anongid=99)
EOF
```

Note

Be sure to replace the directory, network address, and prefix above to match your network. The only space in the line above should be between the directory and the network address.

Systemd Units

Install the NFSv4 server units included in the [blfs-systemd-units-20240801](#) package to start the server at boot.

```
make install-nfsv4-server
```

If you have disabled NFSv4 support, run the following command as the `root` user to omit the NFSv4 specific systemd units:

```
make install-nfs-server
```

You can edit the `/etc/default/nfs-utils` file to change the startup options for NFS daemons. Defaults should be fine for most use cases.

Client Configuration

`/etc/fstab` contains the directories that are to be mounted on the client. Alternately the partitions can be mounted by using the `mount` command with the proper options. To mount the `/home` and `/usr` partitions, add the following to the `/etc/fstab`:

```
<server-name>:/home /home nfs rw,_netdev 0 0  
<server-name>:/usr /usr nfs ro,_netdev 0 0
```

The options which can be used are specified in `man 5 nfs`. If both the client and server are running recent versions of linux, most of the options will be negotiated (but see the Note above on `nfsver=3`). You can specify either `rw` or `ro`, `_netdev` if the filesystem is to be automatically mounted at boot, or `noauto` (and perhaps `user`) for other filesystems.

If the fileserver is not running a recent version of linux, you may need to specify other options.

You may need to enable `autofs v4` in your kernel, and add the option `comment=systemd.automount`. Some machines may need this because `systemd` tries to mount the external filesystems before the network is up. An alternative is to run `mount -a` as the `root` user after the system has started.

Systemd Units

Note

The following `systemd` units are not required if the `nfs-server` units are installed.

Install the `units` included in the [blfs-systemd-units-20240801](#) package to start the client services at boot.

```
make install-nfs-client
```

Contents

Installed Programs: `exportfs`, `fsidd`, `mountstats`, `mount.nfs`, `mount.nfs4` (link to `mount.nfs`), `nfsconf`, `nfsdclnts`, `nfsiostat`, `nfsstat`, `rpc.mountd`, `rpc.nfsd`, `rpc.statd`, `rpcdebug`, `showmount`, `sm-notify`, `start-statd`, `umount.nfs` (link to `mount.nfs`), and `umount.nfs4` (link to `mount.nfs`)

Installed Libraries: None

Installed Directories: `/var/lib/nfs`

Short Descriptions

<code>exportfs</code>	maintains a list of NFS exported file systems
<code>fsidd</code>	offers a local UNIX domain socket interface for all NFS userspace to query the reexport database
<code>mountstats</code>	displays NFS client per-mount statistics
<code>mount.nfs</code>	is used to mount a network share using NFS
<code>mount.nfs4</code>	is used to mount a network share using NFSv4
<code>nfsconf</code>	can be used to test for and retrieve configuration settings from a range of <code>nfs-utils</code> configuration files
<code>nfsdclnts</code>	prints information about NFS clients
<code>nfsiostat</code>	reports input/output statistics for network filesystems
<code>nfsstat</code>	displays statistics kept about NFS client and server activity
<code>rpc.mountd</code>	implements the NFS mount protocol on an NFS server
<code>rpc.nfsd</code>	implements the user level part of the NFS service on the server
<code>rpc.statd</code>	is used by the NFS file locking service. Run on both sides, client as well as server, when you want file locking enabled
<code>rpcedbug</code>	sets or clears the kernel's NFS client and server debug flags

<code>showmount</code>	displays mount information for an NFS server
<code>sm-notify</code>	is used to send Network Status Monitor reboot messages
<code>start-statd</code>	is a script called by nfsmount when mounting a filesystem with locking enabled, if statd does not appear to be running. It can be customised with whatever flags are appropriate for the site
<code>umount.nfs</code>	is used to unmount a network share using NFS
<code>umount.nfs4</code>	is used to unmount a network share using NFSv4

ntp-4.2.8p18

Introduction to ntp

The ntp package contains a client and server to keep the time synchronized between various computers over a network. This package is the official reference implementation of the NTP protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://www.eecis.udel.edu/~ntp/ntp_spool/ntp4/ntp-4.2/ntp-4.2.8p18.tar.gz
- Download MD5 sum: 516bdabd94ab7c824e9771390761a46c
- Download size: 6.8 MB
- Estimated disk space required: 99 MB (with tests)
- Estimated build time: 0.8 SBU (With tests; both using parallelism=4)

ntp Dependencies

Required

[IO-Socket-SSL-2.088](#)

Optional

[libcap-2.70 with PAM](#), [libevent-2.1.12](#), [libedit](#), and [libopts from AutoGen](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/ntp>

Installation of ntp

There should be a dedicated user and group to take control of the `ntpd` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 87 ntp &&
useradd -c "Network Time Protocol" -d /var/lib/ntp -u 87 \
-g ntp -s /bin/false ntp
```

Fix any type issue by executing

```
sed -e "s;pthread_detach(NULL);pthread_detach(0);" \
-i configure \
sntp/configure
```

Install ntp by running the following commands:

```
./configure --prefix=/usr      \
--bindir=/usr/sbin      \
--sysconfdir=/etc      \
--enable-linuxcaps      \
--with-lineeditlibs=readline \
--docdir=/usr/share/doc/ntp-4.2.8p18 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
install -v -o ntp -g ntp -d /var/lib/ntp
```

Command Explanations

--bindir=/usr/sbin: This parameter places the administrative programs in `/usr/sbin`.

--enable-linuxcaps: `ntpd` is run as user `ntp`, so use Linux capabilities for non-root clock control.

--with-lineeditlibs=readline: This switch enables Readline support for `ntpdc` and `ntpq` programs. If omitted, libedit will be used if installed, otherwise no readline capabilities will be compiled.

Configuring ntp

Config Files

`/etc/ntp.conf`

Configuration Information

The following configuration file first defines various `ntp` servers with open access from different continents. Second, it creates a drift file where `ntp` stores the frequency offset and a pid file to store the `ntpd` process ID. Since the documentation included with the package is sparse, visit the `ntp` website at <https://www.ntp.org/> and <https://www.ntppool.org/> for more information.

```
cat > /etc/ntp.conf << "EOF"
# Asia
server 0.asia.pool.ntp.org

# Australia
server 0.oceania.pool.ntp.org

# Europe
server 0.europe.pool.ntp.org

# North America
server 0.north-america.pool.ntp.org

# South America
server 2.south-america.pool.ntp.org

driftfile /var/lib/ntp/ntp.drift
pidfile   /run/ntpd.pid
EOF
```

You may wish to add a “Security session.” For explanations, see <https://www.eecis.udel.edu/~mills/ntp/html/accept.html#restrict>.

```
cat >> /etc/ntp.conf << "EOF"
# Security session
restrict    default limited kod nomodify notrap nopeer noquery
restrict -6 default limited kod nomodify notrap nopeer noquery

restrict 127.0.0.1
restrict ::1
EOF
```

Synchronizing the Time

There are two options. Option one is to run `ntpd` continuously and allow it to synchronize the time in a gradual manner. The other option is to run `ntpd` periodically (using cron) and update the time each time `ntpd` is scheduled.

If you choose Option one, then install the `ntp.service` unit included in the [blfs-systemd-units-20240801](#) package.

```
make install-ntp
```

If you prefer to run `ntpd` periodically, add the following command to `root`'s crontab:

```
ntpd -q
```

Contents

Installed Programs: calc_tickadj, ntp-keygen, ntp-wait, ntpd, ntpdate, ntpdc, ntpq, ntptime, ntptrace, sntp, tickadj, and update-leap

Installed Libraries: None

Installed Directories: /usr/share/ntp, /usr/share/doc/ntp-4.2.8 and /var/lib/ntp

Short Descriptions

<code>calc_tickadj</code>	calculates optimal value for tick given ntp drift file
<code>ntp-keygen</code>	generates cryptographic data files used by the NTPv4 authentication and identification schemes
<code>ntp-wait</code>	is useful at boot time, to delay the boot sequence until <code>ntpd</code> has set the time
<code>ntpd</code>	is a ntp daemon that runs in the background and keeps the date and time synchronized based on response from configured ntp servers. It also functions as a ntp server
<code>ntpdate</code>	is a client program that sets the date and time based on the response from an ntp server. This command is deprecated
<code>ntpdc</code>	is used to query the ntp daemon about its current state and to request changes in that state
<code>ntpq</code>	is a utility program used to monitor <code>ntpd</code> operations and determine performance
<code>ntptime</code>	reads and displays time-related kernel variables
<code>ntptrace</code>	traces a chain of ntp servers back to the primary source
<code>sntp</code>	is a Simple Network Time Protocol (SNTP) client
<code>tickadj</code>	reads, and optionally modifies, several timekeeping-related variables in older kernels that do not have support for precision timekeeping
<code>update-leap</code>	is a script to verify and, if necessary, update the leap-second definition file.

Note

In November 2022, at the 27th General Conference on Weights and Measures, it was decided to abandon the leap second. In addition this script hardcodes a URL for an update file that no longer exists. The last time a leap second was declared was January 2017. This script will probably be removed in a future release.

rpcbind-1.2.7

Introduction to rpcbind

The rpcbind program is a replacement for portmap. It is required for import or export of Network File System (NFS) shared directories.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/rpcbind/rpcbind-1.2.7.tar.bz2>
- Download MD5 sum: acd444ed322eb458fb395ec69c4e083
- Download size: 124 KB
- Estimated disk space required: 1.6 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/rpcbind-1.2.7-vulnerability_fixes-1.patch

rpcbind Dependencies

Required

[libtirpc-1.3.5](#)

Installation of rpcbind

There should be a dedicated user and group to take control of the `rpcbind` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 28 rpc &&
useradd -c "RPC Bind Daemon Owner" -d /dev/null -g rpc \
-s /bin/false -u 28 rpc
```

In order to get `rpcbind` to work properly, first fix the package to use correct service name:

```
sed -i "/servname/s:rpcbind:sunrpc:" src/rpcbind.c
```

Install `rpcbind` by running the following commands:

```
patch -Np1 -i ../rpcbind-1.2.7-vulnerability_fixes-1.patch &&
./configure --prefix=/usr \
--bindir=/usr/sbin \
--enable-warmstarts \
--with-rpcuser=rpc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--with-rpcuser=rpc`: This switch is used so the `rpcbind` daemon will run as an unprivileged user instead of the `root` user.

Configuring `rpcbind`

Systemd Unit

Enable the `systemd` unit installed with the package:

```
systemctl enable rpcbind
```

Contents

Installed Program: `rpcbind` and `rpcinfo`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>rpcbind</code>	is a server that converts RPC program numbers into universal addresses. It must be running on the host to be able to make RPC calls on a server on that machine
<code>rpcinfo</code>	makes an RPC call to an RPC server and reports data according to the requested options

rsync-3.3.0

Introduction to rsync

The `rsync` package contains the `rsync` utility. This is useful for synchronizing large file archives over a network.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.samba.org/ftp/rsync/src/rsync-3.3.0.tar.gz>
- Download MD5 sum: f5c17f9c9164ef9e60d9d8c96b23da06
- Download size: 1.2 MB
- Estimated disk space required: 9.2 MB (with tests; add 24 MB for HTML API documentation)

- Estimated build time: 0.7 SBU (with tests)

rsync Dependencies

Recommended

[popt-1.19](#)

Optional

[Doxygen-1.12.0](#) and [xxhash](#)

Installation of rsync

For security reasons, running the rsync server as an unprivileged user and group is encouraged. If you intend to run `rsync` as a daemon, create the `rsyncd` user and group with the following commands issued by the `root` user:

```
groupadd -g 48 rsyncd &&
useradd -c "rsyncd Daemon" -m -d /home/rsync -g rsyncd \
-s /bin/false -u 48 rsyncd
```

Install rsync by running the following commands:

```
./configure --prefix=/usr      \
--disable-xxhash \
--without-included-zlib &&
make
```

If you have [Doxygen-1.12.0](#) installed and wish to build HTML API documentation, issue:

```
doxygen
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you built the documentation, install it using the following commands as the `root` user:

```
install -v -m755           /usr/share/doc/rsync-3.3.0/api &&
install -v -m644 dox/html/* /usr/share/doc/rsync-3.3.0/api
```

Command Explanations

`--disable-xxhash`: This switch disables advanced xxhash checksum support. Remove this switch if you have installed [xxhash](#).

`--without-included-zlib`: This switch enables compilation with the system-installed zlib library.

Configuring rsync

Config Files

`/etc/rsyncd.conf`

Configuration Information

For client access to remote files, you may need to install the [OpenSSH-9.8p1](#) package to connect to the remote server.

This is a simple download-only configuration to set up running `rsync` as a server. See the [`rsyncd.conf\(5\)`](#) man-page for additional options (i.e., user authentication).

```
cat > /etc/rsyncd.conf << "EOF"
# This is a basic rsync configuration file
# It exports a single module without user authentication.

motd file = /home/rsync/welcome.msg
```

```

use chroot = yes

[localhost]
path = /home/rsync
comment = Default rsync module
read only = yes
list = yes
uid = rsyncd
gid = rsyncd

EOF

```

You can find additional configuration information and general documentation about `rsync` at <https://rsync.samba.org/documentation.html>.

Systemd Unit

Note that you only need to start the rsync server if you want to provide an rsync archive on your local machine. You don't need this unit to run the rsync client.

Install the `rsyncd.service` unit included in the [blfs-systemd-units-20240801](#) package.

```
make install-rsyncd
```

Note

This package comes with two types of units: A service file and a socket file. The service file will start rsync daemon once at boot and it will keep running until the system shuts down. The socket file will make systemd listen on rsync port (Default 873, needs to be edited for anything else) and will start rsync daemon when something tries to connect to that port and stop the daemon when the connection is terminated. This is called socket activation and is analogous to using `{,x}inetd` on a SysVinit based system.

By default, the first method is used - rsync daemon is started at boot and stopped at shutdown. If the socket method is desired, you need to run as the `root` user:

```
systemctl stop rsyncd &&
systemctl disable rsyncd &&
systemctl enable rsyncd.socket &&
systemctl start rsyncd.socket
```

Note that socket method is only useful for remote backups. For local backups you'll need the service method.

Contents

Installed Program: rsync and rsync-ssl

Installed Libraries: None

Installed Directories: Optionally, /usr/share/doc/rsync-3.3.0

Short Descriptions

<code>rsync</code>	is a replacement for <code>rcp</code> (and <code>scp</code>) that has many more features. It uses the "rsync algorithm" which provides a very fast method of syncing remote files. It does this by sending just the differences in the files across the link, without requiring that both sets of files are present at one end of the link beforehand
<code>rsync-ssl</code>	is a helper script used when connecting to an rsync daemon that has SSL support built in

Samba-4.20.4

Introduction to Samba

The Samba package provides file and print services to SMB/CIFS clients and Windows networking to Linux clients. Samba can also be configured as a Windows Domain Controller replacement, a file/print server acting as a member of a Windows Active Directory domain and a NetBIOS (RFC1001/1002) nameserver (which among other things provides LAN browsing support).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.samba.org/pub/samba/stable/samba-4.20.4.tar.gz>
- Download MD5 sum: a366c985dd1b988ff1fda902876f3241
- Download size: 41 MB
- Estimated disk space required: 590 MB (add 64 MB for quicktest)
- Estimated build time: 2.1 SBU (using parallelism=4; add 0.4 SBU for quicktest)

Samba Dependencies

Required

[GnuTLS-3.8.7.1](#), [jansson-2.14](#), [libtirpc-1.3.5](#), [Parse-Yapp-1.21](#), and [rpcsvc-proto-1.4.4](#)

Recommended

[Fuse-3.16.2](#), [PGPME-1.23.2](#), [ICU-75.1](#), [libtasn1-4.19.0](#), [libxslt-1.1.42](#) (for documentation), [Linux-PAM-1.6.1](#), [MIT Kerberos V5-1.21.3](#), and [OpenLDAP-2.6.8](#)

Optional

[Avahi-0.8](#), [BIND-9.20.0](#), [Cups-2.4.10](#), [Cyrus SASL-2.1.28](#), [GDB-15.1](#), [git-2.46.0](#), [GnuPG-2.4.5](#) (required for ADS and the test suite), [libaio-0.3.113](#), [libarchive-3.7.4](#) (for tar in smbclient), [libcap-2.70](#) with PAM, [libgcrypt-1.11.0](#), [libnsl-2.0.1](#), [libunwind-1.6.2](#), [Imdb-0.9.31](#), [Markdown-3.6](#), [nss-3.103](#), [popt-1.19](#), [Talloc-2.4.2](#) (included), [Vala-0.56.17](#), [Valgrind-3.23.0](#) (optionally used by the test suite), [xfsprogs-6.9.0](#), [cmocka](#), [cryptography](#), [ctdb](#) (included), [cwrap](#), [dnspython](#), [FAM](#), [Gamin](#), [GlusterFS](#), [Heimdal](#) (included), [iso8601](#), [ldb](#) (included), [OpenAFS](#), [poetry-core](#) (required for ADS), [pyasn1](#), [tevent](#) (included), [tdb](#) (included), and [tracker-2](#)

Optional (for the Developer Test Suite)

Install in listed order: [six-1.16.0](#), [pytest-8.3.2](#), [argparse](#), [testtools](#), [testscenarios](#), and [python-subunit](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/samba4>

Installation of Samba

To support the test suite, set up a Python virtual environment for some Python modules out of the scope of BLFS:

```
python3 -m venv pyvenv &&
./pyvenv/bin/pip3 install cryptography pyasn1 iso8601
```

Install Samba by running the following commands:

```
PYTHON=$PWD/pyvenv/bin/python3
./configure \
--prefix=/usr \
--sysconfdir=/etc \
--localstatedir=/var \
--with-piddir=/run/samba \
--with-pammodulesdir=/usr/lib/security \
--enable-fhs \
--without-ad-dc \
--with-system-mitkrb5 \
--enable-selftest \
--disable-rpath-install
make
```

To test the results, issue: `PATH=$PWD/pyvenv/bin:$PATH make quicktest`. The test suite will produce lines that look like failures, but these are innocuous. The last few lines of output should report "ALL OK" for a good test run. A summary of any failures can be found in `./st/summary`.

Note

Additionally, developer test suites are available. If you've installed the optional python modules above in the Python virtual environment for building this package, you can run these tests with `make test`. It is not

recommended for the average builder at around 290 SBU and over a gigabyte of disk space, and you should expect ~73 errors and ~30 failures from the 3000+ tests.

Fix hard coded paths to Python 3 interpreter:

```
sed '1s@^.*$@#/usr/bin/python3@' \
-i ./bin/default/source4/scripting/bin/*.inst
```

If upgrading from an old version of samba, as the `root` user, remove the old Python support files to prevent some issues:

```
rm -rf /usr/lib/python3.12/site-packages/samba
```

Still as the `root` user, install the package:

```
make install &&

install -v -m644 examples/smb.conf.default /etc/samba &&

sed -e "s;log file =.*;log file = /var/log/samba/%m.log;" \
-e "s;path = /usr/spool/samba;path = /var/spool/samba;" \
-i /etc/samba/smb.conf.default &&

mkdir -pv /etc/openldap/schema &&

install -v -m644 examples/LDAP/README \
/etc/openldap/schema/README.samba &&

install -v -m644 examples/LDAP/samba* \
/etc/openldap/schema &&

install -v -m755 examples/LDAP/{get*,ol*} \
/etc/openldap/schema
```

Command Explanations

`--enable-fhs`: Assigns all other file paths in a manner compliant with the Filesystem Hierarchy Standard (FHS).

`--without-ad-dc`: Disables Active Directory Domain Controller functionality. See [Set up a Samba Active Directory Domain Controller](#) for detailed information. Remove this switch if you've installed the Python modules needed for ADS support. Note that BLFS does not provide a samba bootscript or systemd unit for an Active Directory domain controller.

`--with-system-mitkrb5`: Enables building with the system version of Kerberos. This mitigates security vulnerabilities and reduces build time. Remove this if you do not have [MIT Kerberos V5-1.21.3](#) installed.

`--disable-rpath-install`: Removes the library installation path from embedded shared library search paths in the installed binary executable files and shared libraries. When this package is installed into the standard location the library installation path is `/usr/lib`. It's always searched by the dynamic linker, so there is no need to embed it into installed files.

`--with-selftest-prefix=SELFTEST_PREFIX`: This option specifies the test suite work directory (default=`./st`).

`install -v -m644 examples/LDAP/* /etc/openldap/schema`: These commands are used to copy sample Samba schemas to the OpenLDAP schema directory.

`install -v -m644 ./examples/smb.conf.default /etc/samba`: This copies a default `smb.conf` file into `/etc/samba`. This sample configuration will not work until you copy it to `/etc/samba/smb.conf` and make the appropriate changes for your installation. See the configuration section for minimum values which must be set.

Configuring Samba

Config Files

`/etc/samba/smb.conf`

Printing to SMB Clients

If you use CUPS for print services, and you wish to print to a printer attached to an SMB client, you need to create an SMB backend device. To create the device, issue the following command as the `root` user:

```
install -dvm 755 /usr/lib/cups/backend &&
ln -v -sf /usr/bin/smbspool /usr/lib/cups/backend/smb
```

Configuration Information

Due to the complexity and the many various uses for Samba, complete configuration for all the package's capabilities is well beyond the scope of the BLFS book. This section provides instructions to configure the `/etc/samba/smb.conf` file for two common scenarios. The complete contents of `/etc/samba/smb.conf` will depend on the purpose of Samba installation.

Note

You may find it easier to copy the configuration parameters shown below into an empty `/etc/samba/smb.conf` file instead of copying and editing the default file as mentioned in the "Command Explanations" section. How you create/edit the `/etc/samba/smb.conf` file will be left up to you. Do ensure the file is only writable by the `root` user (mode 644).

Scenario 1: Minimal Standalone Client-Only Installation

Choose this variant if you only want to transfer files using `smbclient`, mount Windows shares and print to Windows printers, and don't want to share your files and printers to Windows machines.

A `/etc/samba/smb.conf` file with the following three parameters is sufficient:

```
[global]
workgroup = WORKGROUP
dos charset = cp850
unix charset = ISO-8859-1
```

The values in this example specify that the computer belongs to a Windows workgroup named `WORKGROUP`, uses the `cp850` character set on the wire when talking to MS-DOS and MS Windows 9x, and that the filenames are stored in the `ISO-8859-1` encoding on the disk. Adjust these values appropriately for your installation. The `unix charset` value must be the same as the output of `locale charmap` when executed with the `LANG` variable set to your preferred locale, otherwise the `ls` command may not display correct filenames of downloaded files.

There is no need to run any Samba servers in this scenario, thus you don't need to install the provided `systemd` units.

Scenario 2: Standalone File/Print Server

Choose this variant if you want to share your files and printers to Windows machines in your workgroup in addition to the capabilities described in Scenario 1.

In this case, the `/etc/samba/smb.conf.default` file may be a good template to start from. Also, you should add the "dos charset" and "unix charset" parameters to the "[global]" section as described in Scenario 1 in order to prevent filename corruption. For security reasons, you may wish to define `path = /home/alice/shared-files`, assuming your user name is `alice` and you only want to share the files in that directory, instead of your entire home. Then, replace `homes` by `shared-files` and change also the "comment" if used the configuration file below or the `/etc/samba/smb.conf.default` to create yours.

The following configuration file creates a separate share for each user's home directory and also makes all printers available to Windows machines:

```
[global]
workgroup = WORKGROUP
dos charset = cp850
unix charset = ISO-8859-1

[homes]
comment = Home Directories
browseable = no
writable = yes

[printers]
comment = All Printers
path = /var/spool/samba
browseable = no
guest ok = no
printable = yes
```

Other parameters you may wish to customize in the "[global]" section include:

```
server string =
security =
hosts allow =
```

```
load printers =  
log file =  
max log size =  
socket options =  
local master =
```

Reference the comments in the `/etc/samba/smb.conf.default` file for information regarding these parameters.

Since the `smbd` and `nmbd` daemons are needed in this case, install the `samba` **systemd unit**. Be sure to run `smbpasswd` (with the `-a` option to add users) to enable and set passwords for all accounts that need Samba access. Using the default Samba passdb backend, any user you attempt to add will also be required to exist in the `/etc/passwd` file.

Advanced Requirements

More complex scenarios involving domain control or membership are possible. Such setups are advanced topics and cannot be adequately covered in BLFS. Many complete books have been written on these topics alone. Note that in some domain membership scenarios, the `winbindd` daemon and the corresponding **systemd unit** are needed.

Guest account

The default Samba installation uses the `nobody` user for guest access to the server. This can be overridden by setting the `guest account =` parameter in the `/etc/samba/smb.conf` file. If you utilize the `guest account =` parameter, ensure this user exists in the `/etc/passwd` file.

Systemd Units

To start the Samba daemons at boot, install the **systemd units** from the `blfs-systemd-units-20240801` package by running the following command as the `root` user:

```
make install-samba
```

To start the `winbindd` daemon at boot, install the **systemd unit** from the `blfs-systemd-units-20240801` package by running the following command as the `root` user:

```
make install-winbindd
```

Note

This package comes with two types of units: A service file and a socket file. The service file will start the `smbd` daemon once at boot and it will keep running until the system shuts down. The socket file will make **systemd** listen on the `smbd` port (Default 445, needs to be edited for anything else) and will start the `smbd` daemon when something tries to connect to that port and stop the daemon when the connection is terminated. This is called **socket activation** and is analogous to using `{,x}inetd` on a **SysVinit** based system.

By default, the first method is used - the `smbd` daemon is started at boot and stopped at shutdown. If the socket method is desired, you need to run the following commands as the `root` user:

```
systemctl stop smbd &&  
systemctl disable smbd &&  
systemctl enable smbd.socket &&  
systemctl start smbd.socket
```

Note that only the `smbd` daemon can be socket activated.

Contents

Installed Programs: cifsdd, dbwrap_tool, dumpmscat, eventlogadm, gentest, ldbadd, ldbdel, ldbedit, ldbmodify, ldbrename, ldbsearch, locktest, masktest, mdsearch, mvxattr, ndrdump, net, nmbd, nmblookup, ntlm_auth, olschema2ldif, pdredit, profiles, regdiff, regpatch, regshell, regtree, rpcclient, samba-log-parser, samba-gpupdate, samba-regedit, samba-tool, sharesec, smbcacls, smbclient, smbcontrol, smbcquotas, smbd, smbget, smbpasswd, smbspool, smbstatus, smbtar, smbtorture, smbtree, tdbbackup, tdbdump, tdbrestore, tdbtool, testparm, wbinfo, and winbindd

Installed Libraries: libdcerpc-binding.so, libdcerpc-samr.so, libdcerpc-server-core.so, libdcerpc.so, libndr-krb5pac.so, libndr-nbt.so, libndr.so, libndr-standard.so, libnetapi.so, libnss_winbind.so, libnss_wins.so, libsamba-credentials.so, libsamba-errors.so, libsamba-hostconfig.so, libsamba-passdb.so, libsamba-policy.cpython-311-x86_64-linux-gnu.so, libsamba-util.so, libsamdb.so, libsmclient.so, libsmcconf.so, libsmldap.so, libtevent-util.so, libwbclient.so, and filesystem and support modules under `/usr/lib/{python3.12,samba}`

Installed Directories: /etc/samba, /run/samba, /usr/include/samba-4.0, /usr/lib/python3.12/site-packages/samba, /usr/{lib,libexec,share}/samba, and /var/{cache,lib,lock,log,run}/samba

Short Descriptions

cifssdd	is the dd command for SMB
dbwrap_tool	is used to read and manipulate TDB/CTDB databases using the dbwrap interface
dumpmscat	dumps the content of MS catalog files
eventlogadm	is used to write records to eventlogs from STDIN, add the specified source and DLL eventlog registry entries and display the active eventlog names (from <code>smb.conf</code>)
gentest	is used to run random generic SMB operations against two SMB servers and show the differences in behavior
ldbadd	is a command-line utility for adding records to an LDB database
ldbdel	is a command-line utility for deleting LDB database records
ldbedit	allows you to edit LDB databases using your preferred editor
ldbmodify	allows you to modify records in an LDB database
ldbrename	allows you to rename LDB databases
ldbsearch	searches an LDB database for records matching a specified expression
locktest	is used to find differences in locking between two SMB servers
masktest	is used to find differences in wildcard matching between Samba's implementation and that of a remote server
mdsearch	runs Spotlight searches against a SMB server
mvxattr	is used to recursively rename extended attributes
ndrdump	is a DCE/RPC Packet Parser and Dumper
net	is a tool for administration of Samba and remote CIFS servers, similar to the <code>net</code> utility for DOS/Windows
nmbd	is the Samba NetBIOS name server
nmblookup	is used to query NetBIOS names and map them to IP addresses
ntlm_auth	is a tool to allow external access to Winbind's NTLM authentication function
oLschema2ldif	converts LDAP schema's to LDB-compatible LDIF
pdbedit	is a tool used to manage the SAM database
profiles	is a utility that reports and changes SIDs in Windows registry files
regdiff	is a Diff program for Windows registry files
regpatch	applies registry patches to registry files
regshell	is a Windows registry file browser using readline
regtree	is a text-mode registry viewer
rpcclient	is used to execute MS-RPC client side functions
samba-log-parser	parses winbind logs generated by Samba
samba-gpupdate	allows you to edit Microsoft Group Policy Objects (GPOs)
samba-regedit	is a ncurses based tool to manage the Samba registry
samba-tool	is the main Samba administration tool
sharesec	manipulates share ACL permissions on SMB file shares
smbcacls	is used to manipulate Windows NT access control lists
smbclient	is a SMB/CIFS access utility, similar to FTP
smbcontrol	is used to control running <code>smbd</code> , <code>nmbd</code> and <code>winbindd</code> daemons
smbcquotas	is used to manipulate Windows NT quotas on SMB file shares
smbd	is the main Samba daemon which provides SMB/CIFS services to clients
smbget	is a simple utility with <code>wget</code> -like semantics, that can download files from SMB servers. You can specify the files you would like to download on the command-line
smbpasswd	changes a user's Samba password
smbspool	sends a print job to a SMB printer
smbstatus	reports current Samba connections
smbtar	is a shell script used for backing up SMB/CIFS shares directly to Linux tape drives or to a file
smbtorture	is a test suite that runs several tests against a SMB server
smbtree	is a text-based SMB network browser
tdbbackup	is a tool for backing up or validating the integrity of Samba <code>.tdb</code> files

<code>tdbdump</code>	is a tool used to print the contents of a Samba .tdb file
<code>tdbrestore</code>	is a tool for creating a Samba .tdb file out of a ntdbdump
<code>tdbtool</code>	is a tool which allows simple database manipulation from the command line
<code>testparm</code>	checks a <code>smb.conf</code> file for proper syntax
<code>wbinfo</code>	queries a running <code>winbindd</code> daemon
<code>winbindd</code>	resolves names from Windows NT servers
<code>libnss_winbind.so</code>	provides Name Service Switch API functions for resolving names from NT servers
<code>libnss_wins.so</code>	provides API functions for Samba's implementation of the Windows Internet Naming Service
<code>libnetapi.so</code>	provides API functions for the administration tools used for Samba and remote CIFS servers
<code>libsmbclient.so</code>	provides API functions for the Samba SMB client tools
<code>libwbclient.so</code>	provides API functions for Windows domain client services

Wget-1.24.5

Introduction to Wget

The Wget package contains a utility useful for non-interactive downloading of files from the Web.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/wget/wget-1.24.5.tar.gz>
- Download MD5 sum: 271bf949384d0858c2c3d419f6311365
- Download size: 4.9 MB
- Estimated disk space required: 36 MB (add 27 MB for tests)
- Estimated build time: 0.3 SBU (add 0.3 SBU for tests)

Wget Dependencies

Recommended

[libpsl-0.21.5](#)

Recommended at runtime

[make-ca-1.14](#)

Optional

[GnuTLS-3.8.7.1](#), [HTTP-Daemon-6.16](#) (for the test suite), [IO-Socket-SSL-2.088](#) (for the test suite), [libidn2-2.3.7](#), [pcre2-10.44](#), and [Valgrind-3.23.0](#) (for the test suite)

Installation of Wget

Install Wget by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --with-ssl=openssl &&
make
```

To test the results, issue: `make check`.

Some tests may fail when Valgrind tests are enabled.

Now, as the `root` user:

```
make install
```

Command Explanations

--sysconfdir=/etc: This relocates the configuration file from /usr/etc to /etc.
--with-ssl=openssl: This allows the program to use openssl instead of [GnuTLS-3.8.7.1](#).
--enable-valgrind-tests: This allows the tests to be run under valgrind.

Configuring Wget

Config Files

/etc/wgetrc and ~/.wgetrc

Contents

Installed Program: wget

Installed Libraries: None

Installed Directories: None

Short Descriptions

wget retrieves files from the Web using the HTTP, HTTPS and FTP protocols. It is designed to be non-interactive, for background or unattended operations

Configuring the Linux Kernel for Wireless

Before using any userspace tools for connecting to a wireless AP, the Linux kernel must be configured to drive the wireless NIC properly. Enable the following options in the kernel configuration as well as specific device drivers for your hardware and recompile the kernel if necessary:

```
[*] Networking support ---> [NET]
    [*] Wireless ---> [WIRELESS]
        <*/M>   cfg80211 - wireless configuration API [CFG80211]
        < */M> Generic IEEE 802.11 Networking Stack (mac80211) [MAC80211]

Device Drivers --->
    [*] Network device support ---> [NETDEVICES]
        [*] Wireless LAN ---> [WLAN]
```

Open the "Wireless LAN" submenu and select the options that support your hardware. `lspci` from [pciutils-3.13.0](#) or `lsusb` from [usbutils-017](#) can be used to view your hardware configuration. Note that many (though not all) options for the wireless NICs depend on `CONFIG_MAC80211`. After the correct drivers are loaded, the interface will appear in `/sys/class/net`, or in the output of the `ip link` command.

Many wireless NIC drivers require firmware. If you've enabled the correct driver in the kernel configuration but it fails to load (with messages like `Direct firmware load for <filename> failed with error -2`, it means that you need to install the firmware or the wireless NIC won't work. Read [About Firmware](#) for more details.

iw-6.9

Introduction to iw

`iw` is a new nl80211 based CLI configuration utility for wireless devices. It supports all new drivers that have been added to the kernel recently. The old tool `iwconfig`, which uses Wireless Extensions interface, is deprecated and it's strongly recommended to switch to `iw` and `nl80211`.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/software/network/iw/iw-6.9.tar.xz>
- Download MD5 sum: 457c99badf2913bb61a8407ae60e4819
- Download size: 156 KB
- Estimated disk space required: 3.9 MB
- Estimated build time: less than 0.1 SBU

iw Dependencies

Required

[libnl-3.10.0](#)

Required (Runtime)

[Configuring the Linux Kernel for Wireless](#)

Kernel Configuration

To use iw, the kernel must have the appropriate drivers and other support available. Read [Configuring the Linux Kernel for Wireless](#) for details.

Installation of iw

To install iw, use the following commands:

```
sed -i "/INSTALL.*gz/s/.gz//" Makefile &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ...`: Install uncompressed manual pages in accordance with other man pages.

Contents

Installed Programs: iw

Installed Libraries: None

Installed Directories: None

Short Descriptions

`iw` shows / manipulates wireless devices and their configuration

Wireless Tools-29

Introduction to Wireless Tools

The Wireless Extension (WE) is a generic API in the Linux kernel allowing a driver to expose configuration and statistics specific to common Wireless LANs to userspace. A single set of tools can support all the variations of Wireless LANs, regardless of their type, as long as the driver supports Wireless Extensions. WE parameters may also be changed on the fly without restarting the driver (or Linux).

The Wireless Tools (WT) package is a set of tools allowing manipulation of the Wireless Extensions. They use a textual interface to support the full Wireless Extension.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://hewlettpackard.github.io/wireless-tools/wireless_tools.29.tar.gz
- Download MD5 sum: e06c222e186f7cc013fd272d023710cb
- Download size: 288 KB
- Estimated disk space required: 2.0 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/wireless_tools-29-fix_iwlist_scanning-1.patch

Wireless Tools Dependencies

Required (Runtime)

[Configuring the Linux Kernel for Wireless](#)

Kernel Configuration

To use Wireless Tools, the kernel must have the appropriate drivers and other support available. Besides the configurations mentioned by [Configuring the Linux Kernel for Wireless](#), it's also needed to enable the following options in the kernel configuration:

```
[*] Networking support ---> [NET]
[*] Wireless ---> [WIRELESS]
<*/M> cfg80211 - wireless configuration API [CFG80211]
[*]     cfg80211 wireless extensions compatibility [CFG80211_WEXT]
```

Installation of Wireless Tools

First, apply a patch that fixes a problem when numerous networks are available:

```
patch -Np1 -i ../../wireless_tools-29-fix_iwlist_scanning-1.patch
```

To install Wireless Tools, use the following commands:

```
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr INSTALL_MAN=/usr/share/man install
```

Command Explanations

`INSTALL_MAN=/usr/share/man`: Install manual pages in `/usr/share/man` instead of `/usr/man`.

Contents

Installed Programs: ifrename, iwconfig, ievent, iwgetid, iwlist, iwpriv, and iwspy

Installed Library: libiw.so

Installed Directories: None

Short Descriptions

<code>ifrename</code>	renames network interfaces based on various static criteria
<code>iwconfig</code>	configures a wireless network interface
<code>ievent</code>	displays wireless events generated by drivers and setting changes
<code>iwgetid</code>	reports ESSID, NWID or AP/Cell Address of wireless networks
<code>iwlist</code>	gets detailed wireless information from a wireless interface
<code>iwpriv</code>	configures optional (private) parameters of a wireless network interface
<code>iwspy</code>	gets wireless statistics from specific node
<code>libiw.so</code>	contains functions required by the wireless programs and provides an API for other programs

wpa_supplicant-2.11

Introduction to WPA Supplicant

WPA Supplicant is a Wi-Fi Protected Access (WPA) client and IEEE 802.1X supplicant. It implements WPA key negotiation with a WPA Authenticator and Extensible Authentication Protocol (EAP) authentication with an Authentication Server. In

addition, it controls the roaming and IEEE 802.11 authentication/association of the wireless LAN driver. This is useful for connecting to a password protected wireless access point.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://w1.fi/releases/wpa_supplicant-2.11.tar.gz
- Download MD5 sum: 72a4a00eddb7a499a58113c3361ab094
- Download size: 3.7 MB
- Estimated disk space required: 35 MB
- Estimated build time: 0.3 SBU (without optional gui)

WPA Supplicant Dependencies

Required (Runtime)

[Configuring the Linux Kernel for Wireless](#)

Recommended

[desktop-file-utils-0.27](#) (for running `update-desktop-database`) and [libnl-3.10.0](#)

Optional

[libxml2-2.13.3](#), and [qt5-components-5.15.14](#)

Kernel Configuration

To use wpa_supplicant, the kernel must have the appropriate drivers and other support available. Read [Configuring the Linux Kernel for Wireless](#) for details.

Installation of WPA Supplicant

First you will need to create an initial configuration file for the build process. You can read `wpa_supplicant/README` and `wpa_supplicant/defconfig` for the explanation of the following options as well as other options that can be used. Create a build configuration file that should work for standard WiFi setups by running the following command:

```
cat > wpa_supplicant/.config << "EOF"
CONFIG_BACKEND=file
CONFIG_CTRL_IFACE=y
CONFIG_DEBUG_FILE=y
CONFIG_DEBUG_SYSLOG=y
CONFIG_DEBUG_SYSLOG_FACILITY=LOG_DAEMON
CONFIG_DRIVER_NL80211=y
CONFIG_DRIVER_WEXT=y
CONFIG_DRIVER_WIRED=y
CONFIG_EAP_GTC=y
CONFIG_EAP_LEAP=y
CONFIG_EAP_MD5=y
CONFIG_EAP_MSCHAPV2=y
CONFIG_EAP OTP=y
CONFIG_EAP PEAP=y
CONFIG_EAP TLS=y
CONFIG_EAP TTLS=y
CONFIG_IEEE8021X_EAPOL=y
CONFIG_IPV6=y
CONFIG_LIBNL32=y
CONFIG_PEERKEY=y
CONFIG_PKCS12=y
CONFIG_READLINE=y
CONFIG_SMARTCARD=y
CONFIG_WPS=y
CFLAGS += -I/usr/include/libnl3
EOF
```

If you wish to use WPA Supplicant with [NetworkManager-1.48.8](#), make sure that you have installed [dbus-1.14.10](#) and [libxml2-2.13.3](#), then add the following options to the WPA Supplicant build configuration file by running the following command:

```
cat >> wpa_supplicant/.config << "EOF"
CONFIG_CTRL_IFACE_DBUS=y
CONFIG_CTRL_IFACE_DBUS_NEW=y
CONFIG_CTRL_IFACE_DBUS_INTRO=y
EOF
```

Install WPA Suplicant by running the following commands:

```
cd wpa_supplicant &&
make BINDIR=/usr/sbin LIBDIR=/usr/lib
```

If you have installed [qt5-components-5.15.14](#) and wish to build the WPA Suplicant GUI program, run the following commands:

Note

The following directory name is labelled qt4, but is compatible with [qt5-components-5.15.14](#).

```
pushd wpa_gui-qt4 &&
qmake wpa_gui.pro &&
make &&
popd
```

This package does not come with a test suite.

Now, as the `root` user:

```
install -v -m755 wpa_{cli,passphrase,supplicant} /usr/sbin/ &&
install -v -m644 doc/docbook/wpa_supplicant.conf.5 /usr/share/man/man5/ &&
install -v -m644 doc/docbook/wpa_{cli,passphrase,supplicant}.8 /usr/share/man/man8/
```

Install the systemd support files by running the following command as the `root` user:

```
install -v -m644 systemd/*.service /usr/lib/systemd/system
```

If you have built WPA Suplicant with D-Bus support, you will need to install D-Bus configuration files. Install them by running the following commands as the `root` user:

```
install -v -m644 dbus/fi.wl.wpa_supplicant1.service \
/usr/share/dbus-1/system-services/ &&
install -v -d -m755 /etc/dbus-1/system.d &&
install -v -m644 dbus/dbus-wpa_supplicant.conf \
/etc/dbus-1/system.d/wpa_supplicant.conf
```

If you have built the WPA Suplicant GUI program, install it by running the following commands as the `root` user:

```
install -v -m755 wpa_gui-qt4/wpa_gui /usr/bin/ &&
install -v -m644 doc/docbook/wpa_gui.8 /usr/share/man/man8/ &&
install -v -m644 wpa_gui-qt4/wpa_gui.desktop /usr/share/applications/ &&
install -v -m644 wpa_gui-qt4/icons/wpa_gui.svg /usr/share/pixmaps/
```

Note

You will need to restart the system D-Bus daemon before you can use the WPA Suplicant D-Bus interface.

Note

This package installs desktop files into the `/usr/share/applications` hierarchy and you can improve system performance and memory usage by updating `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [desktop-file-utils-0.27](#) installed and issue the following command as the `root` user:

```
update-desktop-database -q
```

Configuring wpa_supplicant

Important

If you are using WPA Suplicant with [NetworkManager-1.48.8](#) (or anything communicating with WPA Suplicant via D-Bus), this section should be skipped. Running a D-Bus connected WPA Suplicant instance and another WPA suplicant instance configured following this section simultaneously can cause subtle issues.

Config File

/etc/wpa_supplicant/wpa_supplicant-*.conf

Configuration Information

To connect to an access point that uses a password, you need to put the pre-shared key in /etc/wpa_supplicant/wpa_supplicant-wifi0.conf. SSID is the string that the access point/router transmits to identify itself. Run the following command as the `root` user:

```
install -v -dm755 /etc/wpa_supplicant &&
wpa_passphrase SSID SECRET_PASSWORD > /etc/wpa_supplicant/wpa_supplicant-wifi0.conf
```

/etc/wpa_supplicant/wpa_supplicant-wifi0.conf can hold the details of several access points. When `wpa_supplicant` is started, it will scan for the SSIDs it can see and choose the appropriate password to connect.

If you want to connect to an access point that isn't password protected, put an entry like this in /etc/wpa_supplicant/wpa_supplicant-wifi0.conf. Replace "Some-SSID" with the SSID of the access point/router.

```
network={
    ssid="Some-SSID"
    key_mgmt=NONE
}
```

Connecting to a new access point that is not in the configuration file can be accomplished manually via the command line or GUI, but it must be done via a privileged user. To do that, add the following to the configuration file:

```
ctrl_interface=DIR=/run/wpa_supplicant GROUP=<privileged group>
update_config=1
```

Replace the <privileged group> above with a system group where members have the ability to connect to a wireless access point.

There are many options that you could use to tweak how you connect to each access point. They are described in some detail in the `wpa_supplicant/wpa_supplicant.conf` file in the source tree.

Connecting to an Access Point

There are 3 types of systemd units that were installed:

- wpa_supplicant@.service
- wpa_supplicant-nl80211@.service
- wpa_supplicant-wired@.service

The only difference between 3 of them is what driver is used for connecting (-D option). The first one uses the default driver, the second one uses the nl80211 driver and the third one uses the wired driver.

You can connect to the wireless access point by running the following command as the `root` user:

```
systemctl start wpa_supplicant@wlan0
```

To connect to the wireless access point at boot, simply enable the appropriate `wpa_supplicant` service by running the following command as the `root` user:

```
systemctl enable wpa_supplicant@wlan0
```

Depending on your setup, you can replace the `wpa_supplicant@.service` with any other listed above.

To assign a network address to your wireless interface, consult the [General Network Configuration](#) page in LFS.

Contents

Installed Programs: wpa_gui, wpa_supplicant, wpa_passphrase and wpa_cli

Installed Libraries: None

Installed Directories: None

Short Descriptions

wpa_gui	is a graphical frontend program for interacting with wpa_supplicant
wpa_supplicant	is a daemon that can connect to a password protected wireless access point
wpa_passphrase	takes an SSID and a password and generates a simple configuration that wpa_supplicant can understand
wpa_cli	is a command line interface used to control a running wpa_supplicant daemon

Chapter 16. Networking Utilities

This chapter contains some tools that come in handy when the network needs investigating.

Avahi-0.8

Introduction to Avahi

The Avahi package is a system which facilitates service discovery on a local network.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lathiat/avahi/releases/download/v0.8/avahi-0.8.tar.gz>
- Download MD5 sum: 229c6aa30674fc43c202b22c5f8c2be7
- Download size: 1.5 MB
- Estimated disk space required: 32 MB
- Estimated build time: 0.3 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/avahi-0.8-ipv6_race_condition_fix-1.patch

Avahi Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended)

Recommended

[GTK+-3.24.43](#), [libdaemon-0.14](#), and [qt5-components-5.15.14](#)

Optional

[D-Bus Python-1.3.2](#), [libevent-2.1.12](#), [Doxygen-1.12.0](#), [GTK+-2](#) (deprecated), and [xmltoman](#) (for generating documentation)

Installation of Avahi

There should be a dedicated user and group to take control of the `avahi-daemon` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -fg 84 avahi &&
useradd -c "Avahi Daemon Owner" -d /run/avahi-daemon -u 84 \
-g avahi -s /bin/false avahi
```

There should also be a dedicated privileged access group for Avahi clients. Issue the following command as the `root` user:

```
groupadd -fg 86 netdev
```

Fix a regression that results in a race condition when IPv6 is in use and multiple network adapters are present on the system:

```
patch -Np1 -i ../avahi-0.8-ipv6_race_condition_fix-1.patch
```

Fix a security vulnerability in `avahi-daemon`:

```
sed -i '426a if (events & AVAHI_WATCH_HUP) { \
client_free(c); \
return; \
}' avahi-daemon/simple-protocol.c
```

Install Avahi by running the following commands:

```
./configure \
--prefix=/usr \
--sysconfdir=/etc \
--localstatedir=/var \
--disable-static \
--disable-libevent \
--disable-mono \
--disable-monodoc \
--disable-python \
--disable-qt3 \
--disable-qt4 \
--enable-core-docs \
--with-distro=none \
--with-dbus-system-address='unix:path=/run/dbus/system_bus_socket' &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-libevent`: This parameter disables the use of [libevent-2.1.12](#). Remove if you have it installed.

`--disable-mono`: This parameter disables the Mono bindings.

`--disable-monodoc`: This parameter disables documentation for the Mono bindings.

`--disable-python`: This parameter disables the scripts that depend on Python. It also allows a regular install to complete successfully.

`--disable-qt3`: This parameter disables the attempt to build the obsolete Qt3 portions of the package.

`--disable-qt4`: This parameter disables the attempt to build the obsolete Qt4Core portions of the package.

`--enable-core-docs`: This parameter enables the building of documentation.

`--with-distro=none`: There is an obsolete boot script in the distribution for LFS. This option disables it.

`--with-dbus-system-address=`: This option prevents the package from referring to the deprecated `/var/run` directory.

`--disable-dbus`: This parameter disables the use of D-Bus.

`--disable-gtk`: This parameter disables the use of GTK+2.

`--disable-gtk3`: This parameter disables the use of GTK+3.

`--disable-qt5`: This parameter disables the use of Qt5, and allows building without it.

`--disable-libdaemon`: This parameter disables the use of libdaemon. If you use this option, `avahi-daemon` won't be built.

`--enable-tests`: This option enables the building of tests and examples.

--enable-compat-howl: This option enables the compatibility layer for HOWL.

--enable-compat-libdns_sd: This option enables the compatibility layer for libdns_sd.

Configuring avahi

Boot Script

To start the `avahi-daemon` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable avahi-daemon
```

To start the `avahi-dnsconfd` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable avahi-dnsconfd
```

Contents

Installed Programs: `avahi-autoipd`, `avahi-browse`, `avahi-browse-domains`, `avahi-daemon`, `avahi-discover-standalone`, `avahi-dnsconfd`, `avahi-publish`, `avahi-publish-address`, `avahi-publish-service`, `avahi-resolve`, `avahi-resolve-address`, `avahi-resolve-host-name`, `avahi-set-host-name`, `bshell`, `bssh`, and `bvnc`

Installed Libraries: `libavahi-client.so`, `libavahi-common.so`, `libavahi-core.so`, `libavahi-glib.so`, `libavahi-gobject.so`, `libavahi-libevent.so`, `libavahi-ui-gtk3.so`, `libavahi-qt5`, `libavahi-ui.so`, `libdns_sd.so`, and `libhowl.so`

Installed Directories: `/etc/avahi/services`, `/usr/include/{avahi-client,avahi-common,avahi-compat-howl,avahi-compat-libdns_sd,avahi-core,avahi-glib,avahi-gobject,avahi-libevent,avahi-qt5,avahi-ui}`, `/usr/lib/avahi`, `/usr/share/avahi`

Short Descriptions

<code>avahi-autoipd</code>	is a IPv4LL network address configuration daemon
<code>avahi-browse</code>	browses for mDNS/DNS-SD services using the Avahi daemon
<code>avahi-browse-domains</code>	browses for mDNS/DNS-SD services using the Avahi daemon
<code>avahi-daemon</code>	is the Avahi mDNS/DNS-SD daemon
<code>avahi-discover-standalone</code>	browses for mDNS/DNS-SD services using the Avahi daemon
<code>avahi-dnsconfd</code>	is a Unicast DNS server from mDNS/DNS-SD configuration daemon
<code>avahi-publish</code>	registers a mDNS/DNS-SD service or host name or address mapping using the Avahi daemon
<code>avahi-publish-address</code>	registers a mDNS/DNS-SD service or host name or address mapping using the Avahi daemon
<code>avahi-publish-service</code>	registers a mDNS/DNS-SD service or host name or address mapping using the Avahi daemon
<code>avahi-resolve</code>	resolves one or more mDNS/DNS host name(s) to IP address(es) (and vice versa) using the Avahi daemon
<code>avahi-resolve-address</code>	resolves one or more mDNS/DNS host name(s) to IP address(es) (and vice versa) using the Avahi daemon
<code>avahi-resolve-host-name</code>	resolves one or more mDNS/DNS host name(s) to IP address(es) (and vice versa) using the Avahi daemon
<code>avahi-set-host-name</code>	changes the mDNS host name
<code>bshell</code>	browses for SSH servers on the local network
<code>bvnc</code>	browses for VNC servers on the local network

BIND Utilities-9.20.0

Introduction to BIND Utilities

BIND Utilities is not a separate package, it is a collection of the client side programs that are included with [BIND-9.20.0](#). The BIND package includes the client side programs `nslookup`, `dig` and `host`. If you install BIND server, these programs will be installed automatically. This section is for those users who don't need the complete BIND server, but need these client side applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.isc.org/isc/bind9/9.20.0/bind-9.20.0.tar.xz>
- Download MD5 sum: 2c8d94d1524cbbae4d76cd74955bb6d9
- Download size: 5.5 MB
- Estimated disk space required: 117 MB
- Estimated build time: 0.5 SBU (using parallelism=4)

BIND Utilities Dependencies

Required

[liburcu-0.14.0](#) and [libuv-1.48.0](#)

Recommended

[JSON-C-0.17](#) and [nghttp2-1.62.1](#)

Optional

[libcap-2.70 with PAM](#), [libxml2-2.13.3](#), and [sphinx-8.0.2](#)

Installation of BIND Utilities

Install BIND Utilities by running the following commands:

```
./configure --prefix=/usr &&
make -C lib/isc    &&
make -C lib/dns    &&
make -C lib/ns     &&
make -C lib/isccfg &&
make -C bin/dig    &&
make -C doc
```

This portion of the package does not come with a test suite.

Now, as the `root` user:

```
make -C lib/isc    install &&
make -C lib/dns    install &&
make -C lib/ns     install &&
make -C lib/isccfg install &&
make -C bin/dig    install &&
cp -v doc/man/{dig.1,host.1,nslookup.1} /usr/share/man/man1
```

Command Explanations

`--disable-doh`: Use this option if you have not installed [nghttp2-1.62.1](#) and you don't need DNS over HTTPS support.

`make -C lib/...`: These commands build the libraries that are needed for the client programs.

`make -C bin/dig`: This command builds the client programs.

`make -C doc`: This command builds the manual pages if the optional Python module [sphinx-8.0.2](#) is installed.

Use `cp -v doc/man/{dig.1,host.1,nslookup.1} /usr/share/man/man1` to install the manual pages if they have been built.

Contents

Installed Programs: dig, host, and nslookup

Installed Libraries: None

Installed Directories: None

Short Descriptions

See the program descriptions in the [BIND-9.20.0](#) section.

NetworkManager-1.48.8

Introduction to NetworkManager

NetworkManager is a set of co-operative tools that make networking simple and straightforward. Whether you use WiFi, wired, 3G, or Bluetooth, NetworkManager allows you to quickly move from one network to another: Once a network has been configured and joined once, it can be detected and re-joined automatically the next time it's available.

This package is known to build and work properly using an LFS 12.2 platform.

Note

Make sure that you disable the `systemd-networkd` service or configure it not to manage the interfaces you want to manage with NetworkManager.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/NetworkManager/1.48/NetworkManager-1.48.8.tar.xz>
- Download MD5 sum: 716942df6afa53f8dad0902f4a3497b4
- Download size: 5.8 MB
- Estimated disk space required: 299 MB (with tests and documentation)
- Estimated build time: 0.9 SBU (with tests, using parallelism=4)

NetworkManager Dependencies

Required

[libndp-1.9](#)

Recommended

[cURL-8.9.1](#), [dhpcd-10.0.8](#), [GLib-2.80.4](#) (with GObject Introspection), [iptables-1.8.10](#), [libpsl-0.21.5](#), [newt-0.52.24](#) (for `nmtui`), [NSS-3.103](#), [Polkit-125](#) (runtime), [PyGObject-3.48.2](#), [Systemd-256.4](#), [Vala-0.56.17](#), and [wpa_supplicant-2.11](#) (runtime, built with D-Bus support)

Optional

[BlueZ-5.77](#), [D-Bus Python-1.3.2](#) (for the test suite), [GnuTLS-3.8.7.1](#) (can be used instead of [NSS-3.103](#)), [GTK-Doc-1.34.0](#), [jansson-2.14](#), [ModemManager-1.18.12](#), [qt5-components-5.15.14](#) with qtdoc (for examples), [UPower-1.90.4](#), [Valgrind-3.23.0](#), [dnsmasq](#), [firewalld](#), [libaudit](#), [libteam](#), [mobile-broadband-provider-info](#), [PPP](#), and [RP-PPPoE](#)

Kernel Configuration

If you wish to run the tests, check that at least the following options are enabled in the kernel configuration. Those options have been determined to be necessary, but may not be sufficient. Recompile the kernel if necessary:

```
[*] Networking support --> [NET]
  Networking options --->
    [*]   TCP/IP networking [INET]
    <*/M>     IP: tunneling [NET_IPIP]
    <*/M>     IP: GRE demultiplexer [NET_IPGRE_DEMUX]
    <*/M>     IP: GRE tunnels over IP [NET_IPGRE]
    <*>       The IPv6 protocol ---> [IPV6]
      <*/M>     IPv6: IPv6-in-IPv4 tunnel (SIT driver) [IPV6_SIT]
      <*/M>     IPv6: GRE tunnel [IPV6_GRE]
      [*]       IPv6: Multiple Routing Tables [IPV6_MULTIPLE_TABLES]
      [*]       MPTCP: Multipath TCP [MPTCP]
      [*]       MPTCP: IPv6 support for Multipath TCP [MPTCP_IPV6]
    <*/M>     802.1Q/802.1ad VLAN Support [VLAN_8021Q]
    [*]       QoS and/or fair queueing --->
      <*>     Stochastic Fairness Queueing (SFQ) [NET_SCH_SFQ]
      <*>     Token Bucket Filter (TBF) [NET_SCH_TBF]
```

```

<*> Fair Queue Controlled Delay AQM (FQ_CODEL)           [NET_SCH_FQ_CODEL]
<*> Ingress/classifier-action Qdisc                      [NET_SCH_INGRESS]

Device Drivers --->
[*] Network device support --->                                [NETDEVICES]
  [*] Network core driver support                               [NET_CORE]
  <*/M> Bonding driver support                                [BONDING]
  <*/M> Dummy net driver support                             [DUMMY]
  <*/M> Ethernet team driver support --->                  [NET_TEAM]
  <*/M> MAC-VLAN support                                    [MACVLAN]
  <*/M> MAC-VLAN based tap driver                           [MACVTAP]
  <*/M> IP-VLAN support                                     [IPVLAN]
  <*/M> Virtual eXtensible Local Area Network (VXLAN)      [VXLAN]
  <*/M> Virtual ethernet pair device                         [VETH]
  <*/M> Virtual Routing and Forwarding (Lite)              [NET_VRF]

```

Installation of NetworkManager

If [qt5-components-5.15.14](#) is installed and the Qt based examples are desired, fix two meson.build files:

```

sed -e 's/-qt4/-qt5/' \
-e 's/moc_location/host_bins/' \
-i examples/C/qt/meson.build &&

sed -e 's/Qt/&5/' \
-i meson.build

```

Fix the python scripts so that they use Python 3:

```
grep -rl '^#!.*python$' | xargs sed -i '1s/python/\&3/'
```

Install NetworkManager by running the following commands:

```

mkdir build &&
cd build &&

CXXFLAGS+=-O2 -fPIC \
meson setup .. \
--prefix=/usr \
--buildtype=release \
-D libaudit=no \
-D nmtui=true \
-D ovs=false \
-D ppp=false \
-D selinux=false \
-D qt=false \
-D session_tracking=systemd \
-D modem_manager=false \
-D docs=true &&
ninja

```

An already active graphical session with a bus address is necessary to run the tests. To test the results, issue `ninja test`.

A few tests may fail, depending on enabled kernel options.

Now, as the `root` user:

```

ninja install &&
mv -v /usr/share/doc/NetworkManager{,-1.48.8}

```

If you have not passed the `-D docs=true` option to `meson`, you can install the pregenerated manual pages with (as the `root` user):

```

for file in $(echo ../man/*.1[578]); do
  section=${file##*.} &&
  install -vdm 755 /usr/share/man/man$section
  install -vm 644 $file /usr/share/man/man$section/
done

```

If you have not used `-D docs=true`, the pregenerated HTML documentation can also be installed with (as the `root` user):

```

cp -Rv ../docs/{api,libnm} /usr/share/doc/NetworkManager-1.48.8

```

Command Explanations

`CXXFLAGS="-O2 -fPIC"`: These compiler options are necessary to build the Qt5 based examples.

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D docs=true`: Use this switch to enable building man pages and documentation if [GTK-Doc-1.34.0](#) is installed.

`-D nmtui=true`: This switch enables building `nmtui`.

`-D ovs=false`: This switch disable the Open vSwitch integration because it needs [jansson-2.14](#). Remove it if you have [jansson-2.14](#) installed on your system.

`-D modem_manager=false`: This switch is required if ModemManager is not installed. Omit this switch if you have built ModemManager and mobile-broadband-provider-info.

`-D session_tracking=systemd`: This switch is used to set `systemd-logind` as the default program for session tracking.

`-D ppp=false`: This switch disables PPP support in NetworkManager since the programs necessary for it are not installed. Remove this switch if you need PPP support and have PPP installed.

`-D libaudit=no` and `-D selinux=false`: This switch disables support for libaudit and SELinux since they are not used in BLFS.

`-D qt=false`: This switch disables the Qt examples. Omit if you have Qt available and wish to install the examples.

`-D crypto=gnutls`: Use this switch if you have GnuTLS installed and want to use it for certificate and key operations in NetworkManager, instead of using NSS (the default).

`-D crypto=null`: Use this switch if neither NSS nor GnuTLS is installed but you want to build NetworkManager anyway. This switch will make NetworkManager lack some features (for example 802.1X).

`-D suspend_resume=upower`: Use this switch if you have [UPower-1.90.4](#) installed and want to use it (instead of [Systemd-256.4](#)) for suspend and resume support.

Configuring NetworkManager

Config Files

`/etc/NetworkManager/NetworkManager.conf`

Configuration Information

For NetworkManager to work, at least a minimal configuration file must be present. Such a file is not installed with `make install`. Issue the following command as the `root` user to create a minimal `NetworkManager.conf` file:

```
cat >> /etc/NetworkManager/NetworkManager.conf << "EOF"
[main]
plugins=keyfile
EOF
```

This file should not be modified directly by users of the system. Instead, system specific changes should be made using configuration files in the `/etc/NetworkManager/conf.d` directory.

To allow polkit to manage authorizations, add the following configuration file:

```
cat > /etc/NetworkManager/conf.d/polkit.conf << "EOF"
[main]
auth-polkit=true
EOF
```

To use something other than the built-in dhcp client (recommended if using only `nmcli`), use the following configuration (valid values include either `dhpcd` or `internal`):

```
cat > /etc/NetworkManager/conf.d/dhcp.conf << "EOF"
[main]
dhcp=dhpcd
EOF
```

To prevent NetworkManager from updating the `/etc/resolv.conf` file, add the following configuration file:

```
cat > /etc/NetworkManager/conf.d/no-dns-update.conf << "EOF"
[main]
dns=none
EOF
```

For additional configuration options, see `man 5 NetworkManager.conf`.

To allow regular users to configure network connections, you should add them to the `netdev` group, and create a polkit rule that grants access. Run the following commands as the `root` user:

```
groupadd -fg 86 netdev &&
/usr/sbin/usermod -a -G netdev <username>

cat > /usr/share/polkit-1/rules.d/org.freedesktop.NetworkManager.rules << "EOF"
polkit.addRule(function(action, subject) {
    if (action.id.indexOf("org.freedesktop.NetworkManager.") == 0 && subject.isInGroup("netdev")) {
        return polkit.Result.YES;
    }
});
EOF
```

Systemd Unit

To start the `NetworkManager` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

Note

If using Network Manager to manage an interface, any previous configuration for that interface should be removed, and the interface brought down prior to starting Network Manager.

```
systemctl enable NetworkManager
```

Starting in version 1.11.2 of NetworkManager, a systemd unit named `NetworkManager-wait-online.service` is enabled, which is used to prevent services that require network connectivity from starting until NetworkManager establishes a connection. To disable this behavior, run the following command as the `root` user:

```
systemctl disable NetworkManager-wait-online
```

Contents

Installed Programs: `NetworkManager`, `nmcli`, `nm-online`, `nmtui`, and, symlinked to `nmtui`: `nmtui-connect`, `nmtui-edit`, and `nmtui-hostname`

Installed Libraries: `libnm.so` and several modules under `/usr/lib/NetworkManager`

Installed Directories: `/etc/NetworkManager`, `/usr/include/libnm`, `/usr/lib/NetworkManager`, `/usr/share/doc/NetworkManager-1.48.8`, `/usr/share/gtk-doc/html/{libnm,NetworkManager}` (if the documentation is built), and `/var/lib/NetworkManager`

Short Descriptions

<code>nmcli</code>	is a command-line tool for controlling NetworkManager and getting its status
<code>nm-online</code>	is an utility to determine whether you are online
<code>nmtui</code>	is an interactive ncurses-based user interface for <code>nmcli</code>
<code>nmtui-connect</code>	is an interactive ncurses-based user interface to activate/deactivate connections
<code>nmtui-edit</code>	is an interactive ncurses-based user interface to edit connections
<code>nmtui-hostname</code>	is an interactive ncurses-based user interface to edit the hostname
<code>NetworkManager</code>	is the network management daemon
<code>libnm.so</code>	contains functions used by NetworkManager

network-manager-applet-1.34.0

Introduction to NetworkManager Applet

The NetworkManager Applet provides a tool and a panel applet used to configure wired and wireless network connections through GUI. It's designed for use with any desktop environment that uses GTK+, such as Xfce and LXDE.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/network-manager-applet/1.34/network-manager-applet-1.34.0.tar.xz>
- Download MD5 sum: 83ff059aff3a691766d5f0079209e5af
- Download size: 1.9 MB
- Estimated disk space required: 46 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

NetworkManager Applet Dependencies

Required

[GTK+-3.24.43](#), [libnma-1.10.6](#), and [libsecret-0.21.4](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [ModemManager-1.18.12](#)

Required (Runtime)

Since this package uses [Polkit-125](#) for authorization, one [Polkit Authentication Agent](#) should be running when the functionality of this package is used.

Optional

[gnome-bluetooth-46.1](#) and [libindicator](#)

Installation of NetworkManager Applet

Install NetworkManager Applet by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D appindicator=no \
    -D selinux=false \
    &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D appindicator=no`: This switch disables AppIndicator support in network-manager-applet because it requires libindicator, which is not in BLFS. The build will fail without this option.

`-D selinux=false`: This switch forcibly disables SELinux support since it is not currently in BLFS and the build will fail without it.

`-D wwan=false`: This switch disables WWAN support. Use this if you do not have [ModemManager-1.18.12](#) installed.

Contents

Installed Programs: `nm-applet` and `nm-connection-editor`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`nm-connection-editor` allows users to view and edit network connection settings

Nmap-7.95

Introduction to Nmap

Nmap is a utility for network exploration and security auditing. It supports ping scanning, port scanning and TCP/IP fingerprinting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://nmap.org/dist/nmap-7.95.tar.bz2>
- Download MD5 sum: b481d293e53b20278d5370458da9a950
- Download size: 11 MB
- Estimated disk space required: 116 MB (add 38 MB for tests)
- Estimated build time: 0.4 SBU (Using parallelism=4; add 0.1 SBU for tests)

Nmap Dependencies

Recommended

Note

These packages are recommended because if they're not installed, the build process will compile and link against its own (often older) version.

[liblinear-247](#), [libpcap-1.10.4](#), [libssh2-1.11.0](#), [Lua-5.4.7](#), [pcre2-10.44](#), and [PyGObject-3.48.2](#)

Optional

[libdnet](#)

Installation of Nmap

First, if it is not already installed, add a needed Python module. As the `root` user:

```
pip3 install build
```

Install Nmap by running the following commands:

```
./configure --prefix=/usr &&
make
```

If you wish to run the test suite, run the following command:

```
sed -e '/import imp/d' \
-e 's/^ndiff = .*$/import ndiff/' \
-i ndiff/ndifftest.py
```

To test the results, issue: `make check` as the `root` user. Tests need a graphical session.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: ncat, ndiff, nmap, nping, uninstall_ndiff, uninstall_zenmap, zenmap, and 2 symlinks to zenmap: nmapfe and xnmap

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/{radialnet,zenmapCore,zenmapGUI,zenmap-7.95-py3.12.egg-info}, and /usr/share/{ncat,nmap,zenmap}

Short Descriptions

ncat	is a utility for reading and writing data across networks from the command line
ndiff	is a tool to aid in the comparison of Nmap scans
nmap	is a utility for network exploration and security auditing. It supports ping scanning, port scanning and TCP/IP fingerprinting
nping	is an open-source tool for network packet generation, response analysis and response time measurement
uninstall_ndiff	is a Python script to uninstall <code>ndiff</code>
uninstall_zenmap	is a Python script to uninstall <code>zenmap</code>
zenmap	is a Python based graphical nmap frontend viewer

Traceroute-2.1.5

Introduction to Traceroute

The Traceroute package contains a program which is used to display the network route that packets take to reach a specified host. This is a standard network troubleshooting tool. If you find yourself unable to connect to another system, traceroute can help pinpoint the problem.

Note

This package overwrites the version of `traceroute` that was installed in the inetutils package in LFS. This version is more powerful and allows many more options than the standard version.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/traceroute/traceroute-2.1.5.tar.gz>
- Download MD5 sum: 6599a83531ecb31275ff7906349c0970
- Download size: 76 KB
- Estimated disk space required: 624 KB
- Estimated build time: less than 0.1 SBU

Installation of Traceroute

Install Traceroute by running the following commands:

```
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make prefix=/usr install          &&
ln -sv -f traceroute /usr/bin/traceroute6      &&
ln -sv -f traceroute.8 /usr/share/man/man8/traceroute6.8 &&
rm -fv /usr/share/man/man1/traceroute.1
```

The traceroute.1 file that was installed in LFS by inetutils is no longer relevant. This package overwrites that version of traceroute and installs the man page in man chapter 8.

Contents

Installed Program: traceroute and traceroute6 (symlink)

Installed Libraries: None

Installed Directories: None

Short Descriptions

- | | |
|--------------------------|--|
| <code>traceroute</code> | does basically what it says: it traces the route your packets take from the host you are working on to another host on a network, showing all the intermediate hops (gateways) along the way |
| <code>traceroute6</code> | is equivalent to <code>traceroute -6</code> |

Whois-5.4.3

Introduction to Whois

Whois is a client-side application which queries the whois directory service for information pertaining to a particular domain name. This package will install two programs by default: `whois` and `mkpasswd`. The `mkpasswd` command is also installed by the `expect` package in LFS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/rfc1036/whois/archive/v5.4.3/whois-5.4.3.tar.gz>
- Download MD5 sum: 381dce8db7c6e38ef013b5d6527f494c
- Download size: 100 KB
- Estimated disk space required: 1.2 MB
- Estimated build time: less than 0.1 SBU

Whois Dependencies

Optional

[libidn-1.42](#) or [libidn2-2.3.7](#)

Installation of Whois

Build the application with:

```
make
```

You can install the `whois` program, the `mkpasswd` program, and the locale files independently. Control your choice of what is installed with the following commands issued as the `root` user:

Note

Installing this version of `mkpasswd` will overwrite the same command installed in LFS.

```
make prefix=/usr install-whois  
make prefix=/usr install-mkpasswd  
make prefix=/usr install-pos
```

Contents

Installed Programs: whois and mkpasswd

Installed Libraries: None

Installed Directories: None

Short Descriptions

- | | |
|--------------------|---|
| <code>whois</code> | is a client-side application which queries the whois directory service for information pertaining to a particular domain name |
|--------------------|---|

`mkpasswd` generates a new password, and optionally applies it to a user

Wireshark-4.2.6

Introduction to Wireshark

The Wireshark package contains a network protocol analyzer, also known as a “sniffer.” This is useful for analyzing data captured “off the wire” from a live network connection, or data read from a capture file.

Wireshark provides both a graphical and a TTY-mode front-end for examining captured network packets from over 500 protocols, as well as the capability to read capture files from many other popular network analyzers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.wireshark.org/download/src/all-versions/wireshark-4.2.6.tar.xz>
- Download MD5 sum: e118da25ca399111a4e5d947385c7c79
- Download size: 43 MB
- Estimated disk space required: 743 MB (171 MB installed)
- Estimated build time: 2.4 SBU (with parallelism=4)

Additional Downloads

- Additional Documentation: <https://www.wireshark.org/download/docs/> (contains links to several different docs in a variety of formats)

Wireshark dependencies

Required

[CMake-3.30.2](#), [c-ares-1.33.0](#), [GLib-2.80.4](#), [libgcrypt-1.11.0](#), [Qt-6.7.2](#), and [Speex-1.2.1](#)

Note

[Qt-6.7.2](#) is not strictly required, since it can be replaced with Qt5. See “Command explanations” below.

Recommended

[libpcap-1.10.4](#) (required to capture data)

Optional

[asciidoctor-2.0.23](#), [Brotli-1.1.0](#), [Doxygen-1.12.0](#), [git-2.46.0](#), [GnuTLS-3.8.7.1](#), [libnl-3.10.0](#), [libxslt-1.1.42](#), [libxml2-2.13.3](#), [Lua-5.2.4](#), [MIT Kerberos V5-1.21.3](#), [nghttp2-1.62.1](#), [qt5-components-5.15.14](#) with qtmultimedia (required if [Qt-6.7.2](#) is not installed), [SBC-2.0](#), [BCG729](#), [liblbc](#), [libsmi](#), [libssh](#), [MaxMindDB](#), [Minizip](#), [Snappy](#), and [Spandsp](#)

Kernel Configuration

The kernel must have the Packet protocol enabled for Wireshark to capture live packets from the network:

```
[*] Networking support -->
    Networking options --->
        <*/M> Packet socket                                [NET]
                                                [PACKET]
```

If built as a module, the name is `af_packet.ko`.

Installation of Wireshark

Wireshark is a very large and complex application. These instructions provide additional security measures to ensure that only trusted users are allowed to view network traffic. First, set up a system group for wireshark. As the `root` user:

```
groupadd -g 62 wireshark
```

Continue to install Wireshark by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D CMAKE_INSTALL_DOCDIR=/usr/share/doc/wireshark-4.2.6 \
      -G Ninja \
      .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install &&

install -v -m755 -d /usr/share/doc/wireshark-4.2.6 &&
install -v -m644    ./README.linux ../doc/README.* ../doc/randpkt.txt \
                  /usr/share/doc/wireshark-4.2.6 &&

pushd /usr/share/doc/wireshark-4.2.6 &&
for FILENAME in ../../wireshark/*.html; do
    ln -s -v -f $FILENAME .
done &&
popd
unset FILENAME
```

If you downloaded any of the documentation files from the page listed in the 'Additional Downloads', install them by issuing the following commands as the `root` user:

```
install -v -m644 <Downloaded_Files> \
          /usr/share/doc/wireshark-4.2.6
```

Now, set ownership and permissions of sensitive applications to only allow authorized users. As the `root` user:

```
chown -v root:wireshark /usr/bin/tshark &&
chmod -v 6550 /usr/bin/tshark
```

Finally, add any users to the wireshark group (as `root` user):

```
usermod -a -G wireshark <username>
```

If you are installing wireshark for the first time, it will be necessary to logout of your session and login again. This will put wireshark in your groups, because otherwise Wireshark will not function properly.

Command Explanations

`-D USE_qt6=OFF`: Use this switch if [Qt-6.7.2](#) is not available. You'll need [qt5-components-5.15.14](#) with qtmultimedia.

Configuring Wireshark

Config Files

`/etc/wireshark.conf` and `~/.config/wireshark/*` (unless there is already `~/.wireshark/*` in the system)

Configuration Information

Though the default configuration parameters are very sane, reference the configuration section of the [Wireshark User's Guide](#) for configuration information. Most of Wireshark's configuration can be accomplished using the menu options of the `wireshark` graphical interfaces.

Note

If you want to look at packets, make sure you don't filter them out with [iptables-1.8.10](#). If you want to exclude certain classes of packets, it is more efficient to do it with iptables than it is with Wireshark.

Contents

- Installed Programs:** capinfos, captypes, editcap, idl2wrs, mergecap, randpkt, rawshark, reordercap, sharkd, text2pcap, tshark, and wireshark
- Installed Libraries:** libwireshark.so, libwiretap.so, libwsutil.so, and numerous modules under /usr/lib/wireshark/plugins
- Installed Directories:** /usr/{lib,share}/wireshark and /usr/share/doc/wireshark-4.2.6

Short Descriptions

<code>capinfos</code>	reads a saved capture file and returns any or all of several statistics about that file. It is able to detect and read any capture supported by the Wireshark package
<code>captypes</code>	prints the file types of capture files
<code>editcap</code>	edits and/or translates the format of capture files. It knows how to read libpcap capture files, including those of <code>tcpdump</code> , Wireshark and other tools that write captures in that format
<code>idl2wrs</code>	is a program that takes a user specified CORBA IDL file and generates "C" source code for a Wireshark "plugin." It relies on two Python programs <code>wireshark_be.py</code> and <code>wireshark_gen.py</code> , which are not installed by default. They have to be copied manually from the <code>tools</code> directory to the <code>\$PYTHONPATH/site-packages/</code> directory
<code>mergecap</code>	combines multiple saved capture files into a single output file
<code>randpkt</code>	creates random-packet capture files
<code>rawshark</code>	dumps and analyzes raw libpcap data
<code>reordercap</code>	reorders timestamps of input file frames into an output file
<code>sharkd</code>	is a daemon that listens on UNIX sockets
<code>text2pcap</code>	reads in an ASCII hex dump and writes the data described into a libpcap -style capture file
<code>tshark</code>	is a TTY-mode network protocol analyzer. It lets you capture packet data from a live network or read packets from a previously saved capture file
<code>wireshark</code>	is the Qt GUI network protocol analyzer. It lets you interactively browse packet data from a live network or from a previously saved capture file
<code>libwireshark.so</code>	contains functions used by the Wireshark programs to perform filtering and packet capturing
<code>libwiretap.so</code>	is a library being developed as a future replacement for <code>libpcap</code> , the current standard Unix library for packet capturing. For more information, see the <code>README</code> file in the source <code>wiretap</code> directory

Chapter 17. Networking Libraries

These applications are support libraries for other applications in the book. It is unlikely that you would just install these libraries, you will generally find that you will be referred to this chapter to satisfy a dependency of other applications.

c-ares-1.33.0

Introduction to c-ares

c-ares is a C library for asynchronous DNS requests.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/c-ares/c-ares/releases/download/v1.33.0/c-ares-1.33.0.tar.gz>
- Download MD5 sum: ff9e46d2ca9fea7159c8939d70b569
- Download size: 1.0 MB
- Estimated disk space required: 11 MB
- Estimated build time: 0.1 SBU

c-ares Dependencies

Required

[CMake-3.30.2](#)

Installation of c-ares

Install c-ares by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr .. &&
make
```

This package does not include a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: adig and ahost

Installed Libraries: libcares.so

Installed Directories: None

Short Descriptions

adig	queries information from DNS servers
ahost	prints the A or AAAA record associated with a hostname or IP address
libcares.so	is a C library for asynchronous DNS requests

cURL-8.9.1

Introduction to cURL

The cURL package contains an utility and a library used for transferring files with URL syntax to any of the following protocols: DICT, FILE, FTP, FTPS, GOPHER, GOPHERS, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, MQTT, POP3, POP3S, RTSP, SMB, SMBS, SMTP, SMPTS, TELNET, and TFTP. Its ability to both download and upload files can be incorporated into other programs to support functions like streaming media.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://curl.se/download/curl-8.9.1.tar.xz>
- Download MD5 sum: 5882f056460240d98cdb5b7f86a06153
- Download size: 2.7 MB
- Estimated disk space required: 42 MB (add 18 MB for tests)
- Estimated build time: 0.2 SBU (with parallelism=4; add 4.9 SBU for tests (without valgrind, add 17 SBU with valgrind))

cURL Dependencies

Recommended

[libpsl-0.21.5](#)

Note

While there is an option to build the package without libpsl, both the upstream developers and the BLFS editors alike highly recommend not disabling support for libpsl due to severe security implications.

Recommended at runtime

[make-ca-1.14](#)

Optional

[Brotli-1.1.0](#), [c-ares-1.33.0](#), [GnuTLS-3.8.7.1](#), [libidn2-2.3.7](#), [libssh2-1.11.0](#), [MIT Kerberos V5-1.21.3](#), [nghttp2-1.62.1](#), [OpenLDAP-2.6.8](#), [Samba-4.20.4](#) (runtime, for NTLM authentication), [gsasl](#), [impacket](#), [libmetalink](#), [librtmp](#), [ngtcp2](#), [quiche](#), and [SPNEGO](#)

Optional if Running the Test Suite

[Apache-2.4.62](#) and [stunnel-5.72](#) (for the HTTPS and FTPS tests), [OpenSSH-9.8p1](#), and [Valgrind-3.23.0](#) (this will slow the tests down and may cause failures)

Installation of cURL

Install cURL by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --with-openssl \
            --enable-threaded-resolver \
            --with-ca-path=/etc/ssl/certs &&
make
```

To run the test suite, issue: `make test`. Some tests are flaky, so if some tests have failed it's possible to run a test again with: `(cd tests; ./runtests.pl <test ID>)` (the ID of failed tests are shown in the "These test cases failed:" message). If you run the tests after the package has been installed, some tests may fail because the man pages were deleted by the 'find' command in the installation instructions below.

Now, as the `root` user:

```
make install &&
rm -rf docs/examples/.deps &&

find docs \( -name Makefile\* -o \
           -name \*.1 -o \
           -name \*.3 -o \
           -name CMakeLists.txt \) -delete &&

cp -v -R docs /usr/share/doc/curl-8.9.1
```

To run some simple verification tests on the newly installed `curl`, issue the following commands: `curl --trace-ascii debugdump.txt https://www.example.com/` and `curl --trace-ascii d.txt --trace-time https://example.com/`. Inspect the locally created trace files `debugdump.txt` and `d.txt`, which contains version information, downloaded files information, etc. One file has the time for each action logged.

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-threaded-resolver`: This switch enables cURL's builtin threaded DNS resolver.

`--with-ca-path=/etc/ssl/certs`: This switch sets the location of the BLFS Certificate Authority store.

`--with-openssl`: This parameter chooses OpenSSL as SSL/TLS implementation. This seems mandatory now.

`--with-gssapi`: This parameter adds Kerberos 5 support to `libcurl`.

`--without-ssl --with-gnutls`: Use this switch to build with GnuTLS support instead of OpenSSL for SSL/TLS.

`--with-ca-bundle=/etc/pki/tls/certs/ca-bundle.crt`: Use this switch instead of `--with-ca-path` if building with GnuTLS support instead of OpenSSL for SSL/TLS.

`--with-libssh2`: This parameter adds SSH support to cURL. This is disabled by default.

`--enable-ares`: This parameter adds support for DNS resolution through the c-ares library. It overrides `--enable-threaded-resolver` and is not widely tested by the editors.

`find docs ... -exec rm {} \;`: This command removes `Makefiles` and `man` files from the documentation directory that would otherwise be installed by the commands that follow.

Contents

Installed Programs: curl and curl-config

Installed Library: libcurl.so

Installed Directories: /usr/include/curl and /usr/share/doc/curl-8.9.1

Short Descriptions

<code>curl</code>	is a command line tool for transferring files with URL syntax
<code>curl-config</code>	prints information about the last compile, like libraries linked to and prefix setting
<code>libcurl.so</code>	provides the API functions required by <code>curl</code> and other programs

GeoClue-2.7.1

Introduction to GeoClue

GeoClue is a modular geoinformation service built on top of the D-Bus messaging system. The goal of the GeoClue project is to make creating location-aware applications as simple as possible.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/geoclue/geoclue/-/archive/2.7.1/geoclue-2.7.1.tar.bz2>
- Download MD5 sum: 08bacd3b45311ee6c20e4240be7bc348
- Download size: 104 KB
- Estimated disk space required: 6.9 MB
- Estimated build time: 0.1 SBU

GeoClue Dependencies

Required

[JSON-GLib-1.8.0](#) and [libsoup-3.4.4](#)

Recommended

[Avahi-0.8](#), [libnotify-0.8.3](#), [ModemManager-1.18.12](#), and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of GeoClue

Install GeoClue by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk-doc=false   \
            ...
            &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D 3g-source=false`: This switch disables the 3G backend. Use it if you have not installed the ModemManager package.

-D modem-gps-source=false: This switch disables the modem GPS backend. Use it if you have not installed the ModemManager package.

-D cdma-source=false: This switch disables the CDMA source backend. Use it if you have not installed the ModemManager package.

-D nmea-source=false: This switch disables the NMEA source. Use it if you have not installed the Avahi package.

-D demo-agent=false: This switch disables the demo. Use it if you have not installed the libnotify package.

Configuring GeoClue

Config Files

/etc/geoclue/conf.d/90-lfs-google.conf

Configuration Information

In March of 2024, Mozilla announced the shutdown of the Mozilla Location Service. Geoclue uses this service for determining a user's location when requested by other applications. The only supported alternative by upstream is to use Google's Geolocation Service.

To use Google's Geolocation Service, an API key must be used, and a configuration file must be created. **This API key is only intended for use with LFS. Please do not use this API key if you are building for another distro or distributing binary copies. If you need an API key, you can request one at <https://www.chromium.org/developers/how-tos/api-keys>.**

Create the configuration needed for using Google's Geolocation Service as the `root` user:

```
cat > /etc/geoclue/conf.d/90-lfs-google.conf << "EOF"
# Begin /etc/geoclue/conf.d/90-lfs-google.conf

# This configuration applies for the WiFi source.
[wifi]

# Set the URL to Google's Geolocation Service.
url=https://www.googleapis.com/geolocation/v1/geolocate?key=AIzaSyDxKL42zsPjbke5O8_rPVpVrLrJ8aeE9rQ

# End /etc/geoclue/conf.d/90-lfs-google.conf
EOF
```

If you do not wish to request your location from a Geolocation service, you can hardcode your location in `/etc/geolocation` using the format described in [geoclue\(5\)](#).

Contents

Installed Programs: None

Installed Libraries: libgeoclue-2.so

Installed Directories: /etc/geoclue, /usr/include/libgeoclue-2.0, /usr/libexec/geoclue-2.0, and /usr/share/gtk-doc/html/{geoclue,libgeoclue}

glib-networking-2.80.0

Introduction to GLib Networking

The GLib Networking package contains Network related gio modules for GLib.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/glib-networking/2.80/glib-networking-2.80.0.tar.xz>
- Download MD5 sum: 8e13b80148e28adc9e4cf681c0b30402
- Download size: 284 KB
- Estimated disk space required: 5.6 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

GLib Networking Dependencies

Required

[GLib-2.80.4](#) and [GnuTLS-3.8.7.1](#)

Recommended

[gsettings-desktop-schemas-46.1](#) (for the applications using this package to use proxy server settings in GNOME) and [make-ca-1.14](#)

Optional

[libproxy](#)

Installation of GLib Networking

Install GLib Networking by running the following commands:

```
mkdir build &&
cd build &&

meson setup \
    --prefix=/usr \
    --buildtype=release \
    -D libproxy=disabled \
    ... &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: None

Installed Libraries: libgiognomeproxy.so and libgiognutls.so (GIO Modules installed in `/usr/lib/gio/modules`)

Installed Directories: None

kdsoap-2.2.0

Introduction to kdsoap

The kdsoap is Qt-based client-side and server-side SOAP component. It can be used to create client applications for web services and also provides the means to create web services without the need for any further component such as a dedicated web server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KDAB/KDSoap/releases/download/kdsoap-2.2.0/kdsoap-2.2.0.tar.gz>
- Download MD5 sum: a4ef201402aaa1500439a2ed4359c0f3
- Download size: 11 MB
- Estimated disk space required: 49 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

kdsoap Dependencies

Required

[qt5-components-5.15.14](#)

Installation of kdsoap

Install kdsoap by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_DOCDIR=/usr/share/doc/kdsoap-2.2.0 \
.. &&
make
```

The package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: kdwsdl2cpp

Installed Libraries: libkdsoap.so and libkdsoap-server.so

Installed Directories: /usr/lib/cmake/KDSOap, /usr/share/doc/kdsoap-2.2.0, /usr/include/KDSOapClient. and /usr/include/KDSOapServer

Idns-1.8.4

Introduction to Idns

Idns is a fast DNS library with the goal to simplify DNS programming and to allow developers to easily create software conforming to current RFCs and Internet drafts. This packages also includes the `drill` tool.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.nlnetlabs.nl/downloads/Idns/Idns-1.8.4.tar.gz>
- Download MD5 sum: 94ea1ed8fc7095bef003b64b53b71ec7
- Download size: 1.2 MB
- Estimated disk space required: 31 MB (with docs)
- Estimated build time: 0.2 SBU (with docs)

Idns Dependencies

Optional

[make-ca-1.14](#) and [libpcap-1.10.4](#) (for example programs), [SWIG-4.2.1](#) (for Python bindings), and [Doxygen-1.12.0](#) (for html documentation)

Installation of Idns

Install Idns by running the following commands:

```
./configure --prefix=/usr      \
--sysconfdir=/etc      \
--disable-static      \
--with-drill        &&
make
```

If you have [Doxygen-1.12.0](#) installed and want to build html documentation, run the following command:

```
make doc
```

This package does not come with a working test suite.

Now, as the `root` user:

```
make install
```

If you built html documentation, install it by running the following commands as the `root` user:

```
install -v -m755 -d /usr/share/doc/ldns-1.8.4 &&
install -v -m644 doc/html/* /usr/share/doc/ldns-1.8.4
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-drill`: This option enables building of the `drill` tool (used for obtaining debug information from DNS(SEC))

`--disable-dane-ta-usage`: This option disables DANE-TA (DNS-Based Authentication of Named Entities) support. It is only needed if OpenSSL-1.1.0 or later is not installed.

`--with-examples`: This option enables building of the example programs.

`--with-pyldns`: This option enables building of the Python bindings. If you use this option, you will need to add `PYTHON=/usr/bin/python3` to the configure line as well.

Contents

Installed Programs: `drill` and `ldns-config`

Installed Library: `libldns.so`

Installed Directories: `/usr/include/ldns` and `/usr/share/doc/ldns-1.8.4`

Short Descriptions

<code>drill</code>	is a tool like <code>dig</code> from BIND Utilities-9.20.0 designed to get all sorts of information out of the DNS
<code>ldns-config</code>	shows compiler and linker flags for ldns usage
<code>libldns.so</code>	provides the ldns API functions to programs

libevent-2.1.12

Introduction to libevent

libevent is an asynchronous event notification software library. The libevent API provides a mechanism to execute a callback function when a specific event occurs on a file descriptor or after a timeout has been reached. Furthermore, libevent also supports callbacks due to signals or regular timeouts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libevent/libevent/releases/download/release-2.1.12-stable/libevent-2.1.12-stable.tar.gz>
- Download MD5 sum: b5333f021f880fe76490d8a799cd79f4
- Download size: 1.0 MB
- Estimated disk space required: 20 MB (add 4 MB for tests and 4 MB for API docs)
- Estimated build time: 0.3 SBU (add 11 SBU for tests)

libevent Dependencies

Optional

[Doxygen-1.12.0](#) (for API documentation)

Installation of libevent

First, fix an issue that prevents event_rpcgen.py from working:

```
sed -i 's/python/&3/' event_rpcgen.py
```

Install libevent by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

If you have [Doxygen-1.12.0](#) installed and wish to build API documentation, issue :

```
doxygen Doxyfile
```

To test the results, issue: `make verify`. Six tests in every suite related to `regress_ssl.c` and `regress_http.c` are known to fail due to incompatibilities with OpenSSL-3. Some tests that are related to `regress_dns.c` are also known to fail intermittently due to insufficient test timeouts.

Now, as the `root` user:

```
make install
```

If you built the API documentation, install it by issuing the following commands as the `root` user:

```
install -v -m755 -d /usr/share/doc/libevent-2.1.12/api &&  
cp -v -R doxygen/html/* \  
/usr/share/doc/libevent-2.1.12/api
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: event_rpcgen.py

Installed Libraries: libevent_core.so, libevent_extra.so, libevent_openssl.so, libevent_pthreads.so and libevent.so

Installed Directory: /usr/include/event2 and /usr/share/doc/libevent-2.1.12

libmnl-1.0.5

Introduction to libmnl

The libmnl library provides a minimalistic userspace library oriented to Netlink developers. There are a lot of common tasks in the parsing, validating, and constructing of both the Netlink header and TLVs that are repetitive and easy to get wrong. This library aims to provide simple helpers that allow you to re-use code and to avoid re-inventing the wheel.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://netfilter.org/projects/libmnl/files/libmnl-1.0.5.tar.bz2>
- Download MD5 sum: 0bbb70573119ec5d49435114583e7a49
- Download size: 308 KB
- Estimated disk space required: 2.9 MB
- Estimated build time: less than 0.1 SBU

libnma Dependencies

Optional

[Doxygen-1.12.0](#)

Installation of libmnl

Install libmnl by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Library: libmnl.so

Short Descriptions

libmnl.so provides functions for parsing, validating, constructing of both the Netlink header and TLVs

libndp-1.9

Introduction to libndp

The libndp package provides a wrapper for IPv6 Neighbor Discovery Protocol. It also provides a tool named ndptool for sending and receiving NDP messages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://libndp.org/files/libndp-1.9.tar.gz>
- Download MD5 sum: 9d486750569e7025e5d0afdcc509b93c
- Download size: 368 KB
- Estimated disk space required: 2.5 MB
- Estimated build time: less than 0.1 SBU

Installation of libndp

Install libndp by running the following command:

```
./configure --prefix=/usr      \  
          --sysconfdir=/etc      \  
          --localstatedir=/var \  
          --disable-static      &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: ndptool

Installed Library: libndp.so

Installed Directory: None

Short Descriptions

ndptool is a tool for sending and receiving NDP messages

libndp.so provides a wrapper for the IPv6 Neighbor Discovery Protocol

libnl-3.10.0

Introduction to libnl

The libnl suite is a collection of libraries providing APIs to netlink protocol based Linux kernel interfaces.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/thom311/libnl/releases/download/libnl3_10_0/libnl-3.10.0.tar.gz
 - Download MD5 sum: 504f3929a3d878fcaccc8a19f1a5f449
 - Download size: 1.1 MB
 - Estimated disk space required: 32 MB (with API documentation and tests)
 - Estimated build time: 0.5 SBU (with API documentation and tests)

Optional Download

- Download (HTTP): https://github.com/thom311/libnl/releases/download/libnl3_10_0/libnl-doc-3.10.0.tar.gz
 - Download MD5 sum: ebbb10af7dafac18640a0974d255d2b7
 - Download size: 3.8 MB

Installation of libnl

Install libnl by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --disable-static  &&
make
```

Note

If the `make` command was executed with multiple parallel jobs enabled, it might disrupt the terminal mode and cause some “amusing” visual effects. In the worst case, this issue may cause keyboard inputs not displayed on the screen at all (but you can still run any command if you can type it correctly). Run the `reset` to fix such an issue.

If you wish to run the tests, check that the following options are enabled in the kernel configuration and recompile the kernel if necessary. Some of them may not be strictly needed, but they should support a complete test coverage.

```
General setup --->
  -- Namespaces support --->
    [*] User namespace                                [NAMESPACES]
    [*] Network namespace                            [USER_NS]
    [*] Network namespaces                           [NET_NS]

  [*] Networking support --->
    Networking options --->
      [*] TCP/IP networking                         [INET]
      [*] IP: advanced router                      [IP_ADVANCED_ROUTER]
      [*] IP: policy routing                       [IP_MULTIPLE_TABLES]
      <*/M> IP: tunneling                          [NET_IPPIP]
      <*/M> IP: GRE demultiplexer                  [NET_IPGRE_DEMUX]
      <*/M> IP: GRE tunnels over IP                [NET_IPGRE]
      <*/M> Virtual (secure) IP: tunneling        [NET_IPVTI]
      <*> The IPv6 protocol --->
        <*/M> IPv6: IPv6-in-IPv4 tunnel (SIT driver) [IPV6_SIT]
        <*/M> IPv6: IP-in-IPv6 tunnel (RFC2473)     [IPV6_TUNNEL]
        [*] IPv6: Multiple Routing Tables          [IPV6_MULTIPLE_TABLES]
    [*] Network packet filtering framework (Netfilter) ---> [NETFILTER]
      Core Netfilter Configuration --->
        <*/M> Netfilter nf_tables support           [NF_TABLES]
        [*] Netfilter nf_tables netdev tables support [NF_TABLES_NETDEV]
        {*/M} Netfilter packet duplication support   [NF_DUP_NETDEV]
        <*/M> Netfilter nf_tables netdev packet forwarding support [NET_FWD_NETDEV]
```

```

<*/M> 802.1d Ethernet Bridging [BRIDGE]
<*/M> 802.1Q/802.1ad VLAN Support [VLAN_8021Q]
--*- L3 Master device support [NET_L3_MASTER_DEV]

Device Drivers --->
[*] Network device support ---> [NETDEVICES]
[*] Network core driver support [NET_CORE]
<*/M> Bonding driver support [BONDING]
<*/M> Dummy net driver support [DUMMY]
<*/M> Intermediate Functional Block support [IFB]
<*/M> MAC-VLAN support [MACVLAN]
<*/M> MAC-VLAN based tap driver [MACVTAP]
<*/M> IP-VLAN support [IPVLAN]
<*/M> Virtual eXtensible Local Area Network (VXLAN) [VXLAN]
<*/M> IEEE 802.1AE MAC-level encryption (MACsec) [MACSEC]
<*/M> Virtual ethernet pair device [VETH]
<*/M> Virtual Routing and Forwarding (Lite) [NET_VRF]

```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you wish to install the API documentation, as the `root` user:

```
mkdir -vp /usr/share/doc/libnl-3.10.0 &&
tar -xf ./libnl-doc-3.10.0.tar.gz --strip-components=1 --no-same-owner \
-C /usr/share/doc/libnl-3.10.0
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-cli`: Use this parameter if you don't want to install cli tools provided by the package.

Contents

Installed Programs: genl-ctrl-list, idiag-socket-details, nl-class-add, nl-class-delete, nl-classid-lookup, nl-class-list, nl-cls-add, nl-cls-delete, nl-cls-list, nl-link-list, nl-pktloc-lookup, nl-qdisc-add, nl-qdisc-delete, nl-qdisc-list, and 48 other helper programs with nl- and nf- prefixes

Installed Libraries: libnl-3.so, libnl-cli-3.so, libnl-genl-3.so, libnl-idiag-3.so, libnl-nf-3.so, libnl-route-3.so, libnl-xfrm-3.so, and cli modules under /usr/lib/libnl/cli tree

Installed Directories: /etc/libnl, /usr/include/libnl3, /usr/lib/libnl, and /usr/share/doc/libnl-3.10.0

Short Descriptions

<code>genl-ctrl-list</code>	queries the Generic Netlink controller in the kernel and prints a list of all registered Generic Netlink families including the version of the interface that has been registered
<code>nl-class-add</code>	adds, updates, or replaces Traffic Classes
<code>nl-class-delete</code>	deletes Traffic Classes
<code>nl-classid-lookup</code>	is used to resolve qdisc/class names to classid values and vice versa
<code>nl-class-list</code>	lists Traffic Classes
<code>nl-cls-add</code>	adds a classifier
<code>nl-cls-delete</code>	deletes a classifier
<code>nl-cls-list</code>	lists classifiers
<code>nl-link-list</code>	dumps link attributes
<code>nl-pktloc-lookup</code>	allows the lookup of packet location definitions
<code>nl-qdisc-add</code>	adds queueing disciplines (qdiscs) in the kernel
<code>nl-qdisc-delete</code>	deletes queueing disciplines (qdiscs) in the kernel
<code>nl-qdisc-list</code>	lists queueing disciplines (qdiscs) in the kernel
<code>libnl*-3.so</code>	These libraries contain API functions used to access Netlink interfaces in Linux kernel

libnma-1.10.6

Introduction to libnma

The libnma package contains an implementation of the NetworkManager GUI functions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libnma/1.10/libnma-1.10.6.tar.xz>
- Download MD5 sum: 71c7ce674fea1fae8f1368a7fcb6ff43
- Download size: 688 KB
- Estimated disk space required: 15 MB
- Estimated build time: 0.1 SBU (Using parallelism=4; with test)

libnma Dependencies

Required

[Gcr-3.41.2](#), [GTK+-3.24.43](#), [ISO Codes-4.16.0](#), and [NetworkManager-1.48.8](#)

Recommended

[GTK-4.14.5](#) and [Vala-0.56.17](#)

Optional

[mobile-broadband-provider-info](#)

Installation of libnma

Install libnma by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gtk_doc=false \
    -D libnma_gtk4=true \
    -D mobile_broadband_provider_info=false &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: This parameter disables generating API documentation using gtk-doc. If you have [GTK-Doc-1.34.0](#) installed and wish to generate the API documentation, omit this switch.

`-D libnma_gtk4=true`: This parameter builds the GTK-4 version of libnma. It is needed by [gnome-control-center-46.4](#).

`-D mobile_broadband_provider_info=false`: This switch disables WWAN support because mobile-broadband-provider-info is not in BLFS. Omit this switch if you have it installed and wish to enable WWAN support.

Contents

Installed Programs: None

Installed Libraries: libnma.so and libnma-gtk4.so

Installed Directories: /usr/include/libnma

Short Descriptions

libnma.so	contains the NetworkManager GUI library
libnma-gtk4.so	contains the GTK-4 version of the NetworkManager GUI library

libnsl-2.0.1

Introduction to libnsl

The libnsl package contains the public client interface for NIS(YP). It replaces the NIS library that used to be in glibc.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/thkukuk/libnsl/releases/download/v2.0.1/libnsl-2.0.1.tar.xz>
- Download MD5 sum: fb178645dfa85ebab0f1e42e219b42ae
- Download size: 276 KB
- Estimated disk space required: 3.1 MB
- Estimated build time: less than 0.1 SBU

libnsl Dependencies

Required

[libtirpc-1.3.5](#)

Installation of libnsl

Install libnsl by running the following commands:

```
./configure --sysconfdir=/etc --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libnsl.so

Installed Directories: /usr/include/rpcsvc

Short Descriptions

libnsl.so	provides the NIS (YP) API functions required by other programs
-----------	--

libpcap-1.10.4

Introduction to libpcap

libpcap provides functions for user-level packet capture, used in low-level network monitoring.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.tcpdump.org/release/libpcap-1.10.4.tar.gz>
- Download MD5 sum: 0322e28dd76cda8066bb6d00fee5969b
- Download size: 932 KB
- Estimated disk space required: 9.9 MB
- Estimated build time: less than 0.1 SBU

libpcap Dependencies

Optional

[BlueZ-5.77](#), [libnl-3.10.0](#), [libusb-1.0.27](#), Software distribution for the [DAG](#), and [Septel](#) range of passive network monitoring cards.

Installation of libpcap

Install libpcap by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

If you want to disable installing the static library, use this sed:

```
sed -i '/INSTALL_DATA.*libpcap.a\|RANLIB.*libpcap.a/ s/^/#/' Makefile
```

Now, as the `root` user:

```
make install
```

Contents

Installed Program: pcap-config

Installed Libraries: libpcap.so

Installed Directory: /usr/include/pcap

Short Descriptions

`pcap-config` provides configuration information for libpcap
`libpcap.{a,so}` are libraries used for user-level packet capture

libpsl-0.21.5

Introduction to libpsl

The libpsl package provides a library for accessing and resolving information from the Public Suffix List (PSL). The PSL is a set of domain names beyond the standard suffixes, such as .com.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/rockdaboot/libpsl/releases/download/0.21.5/libpsl-0.21.5.tar.gz>
- Download MD5 sum: 870a798ee9860b6e77896548428dba7b
- Download size: 7.3 MB
- Estimated disk space required: 50 MB
- Estimated build time: less than 0.1 SBU (including tests)

libpsl Dependencies

Required

[libidn2-2.3.7](#) and [libunistring-1.2](#)

Optional

[GTK-Doc-1.34.0](#) (for documentation) and [Valgrind-3.23.0](#) (for tests)

Installation of libpsl

Install libpsl by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release &&

ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: psl

Installed Library: libpsl.so

Installed Directories: None

Short Descriptions

<code>psl</code>	queries the Public Suffix List
<code>libpsl.so</code>	contains a library used to access the Public Suffix List

libslirp-4.8.0

Introduction to libslirp

Libslirp is a user-mode networking library used by virtual machines, containers or various tools.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/slirp/libslirp/-/archive/v4.8.0/libslirp-v4.8.0.tar.bz2>
- Download MD5 sum: 975605bcc503cc092b6a01351b927a60
- Download size: 126 KB
- Estimated disk space required: 3.1 MB
- Estimated build time: less than 0.1 SBU

libslirp Dependencies

Required

[GLib-2.80.4](#)

Installation of Libslirp

Install libslirp by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&

ninja
```

To test the results, issue: `ninja test`

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: None

Installed Library: libslirp.so

Installed Directory: /usr/include/slirp

Short Descriptions

`libclirp.so` contains user-mode TCP-IP emulation functions

libsoup-2.74.3

Introduction to libsoup

The libsoup is a HTTP client/server library for GNOME. It uses GObject and the GLib main loop to integrate with GNOME applications and it also has an asynchronous API for use in threaded applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libsoup/2.74/libsoup-2.74.3.tar.xz>
- Download MD5 sum: 8f657fd301a213629204b3320c35d75a
- Download size: 1.4 MB
- Estimated disk space required: 17 MB (with tests)
- Estimated build time: 0.3 SBU (Using parallelism=4; with tests)

libsoup Dependencies

Required

[glib-networking-2.80.0](#), [libpsl-0.21.5](#), [libxml2-2.13.3](#), and [SQLite-3.46.1](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[Apache-2.4.62](#) (required to run the test suite), [Brotli-1.1.0](#), [cURL-8.9.1](#) (required to run the test suite), [MIT Kerberos V5-1.21.3](#) (required to run the test suite), [GTK-Doc-1.34.0](#), [PHP-8.3.10](#) compiled with XMLRPC-EPI support (only used for the XMLRPC regression tests), [Samba-4.20.4](#) (ntlm_auth is required to run the test suite), and [sysprof](#)

Installation of libsoup

Install libsoup by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D vapi=enabled    \
            -D gssapi=disabled \
            -D sysprof=disabled \
            ..
            &&
ninja
```

To test the results, issue: `ninja test`. One test named `ssl-test` is known to fail.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D vapi=disabled`: Use this if you have not installed Vala, e.g. because you are not building GNOME.

`-D doc=enabled`: Use this option if you want to build the documentation. Note that you must have [GTK-Doc-1.34.0](#) installed.

`-D gssapi=disabled`: libsoup defaults to building with GSSAPI support, which requires Kerberos (as does the test suite). If you are building GNOME or have kerberos installed, remove this option.

`-D sysprof=disabled`: libsoup will automatically download a git version of [sysprof](#) if git is available or use the installed version if it is present on the system. If you need profiling, remove this option.

Contents

Installed Programs: None

Installed Libraries: libsoup-2.4.so and libsoup-gnome-2.4.so

Installed Directories: /usr/include/libsoup-2.4, /usr/include/libsoup-gnome-2.4 and /usr/share/gtk-doc/html/libsoup-2.4

Short Descriptions

<code>libsoup-2.4.so</code>	provides functions for asynchronous HTTP connections
<code>libsoup-gnome-2.4.so</code>	provides GNOME specific features

libsoup-3.4.4

Introduction to libsoup3

The libsoup3 is a HTTP client/server library for GNOME. It uses GObject and the GLib main loop to integrate with GNOME applications and it also has an asynchronous API for use in threaded applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libsoup/3.4/libsoup-3.4.4.tar.xz>
- Download MD5 sum: a63ea04a9686e9e4470b127ffe1eb96b
- Download size: 1.5 MB
- Estimated disk space required: 38 MB (with tests)

- Estimated build time: 0.3 SBU (Using parallelism=4; with tests)

libsoup3 Dependencies

Required

[glib-networking-2.80.0](#), [libpsl-0.21.5](#), [libxml2-2.13.3](#), [nghttp2-1.62.1](#), and [SQLite-3.46.1](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[Apache-2.4.62](#) (required to run the test suite), [Brotli-1.1.0](#), [cURL-8.9.1](#) (required to run the test suite), [Gi-DocGen-2024.1](#), [MIT Kerberos V5-1.21.3](#) (required to run the test suite), [PHP-8.3.10](#) compiled with XMLRPC-EPI support (only used for the XMLRPC regression tests), [Samba-4.20.4](#) (ntlm_auth is required to run the test suite), [sysprof](#), and [wstest](#)

Installation of libsoup3

Fix the installation path of API documentation:

```
sed 's/apiversion/soup_version/' -i docs/reference/meson.build
```

Install libsoup3 by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D vapi=enabled \
            -D gssapi=disabled \
            -D sysprof=disabled \
            --wrap-mode=nofallback \
            ..
            &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`--wrap-mode=nofallback`: This switch prevents `meson` from using subproject fallbacks for any dependency declarations in the build files, stopping it downloading any optional dependency which is not installed on the system.

`-D vapi=disabled`: Use this if you have not installed Vala, e.g. because you are not building GNOME.

`-D docs=enabled`: If [Gi-DocGen-2024.1](#) is installed, the API documentation will be built and installed no matter if this option is used or not. This option causes the `meson` command to fail if [Gi-DocGen-2024.1](#) is not installed.

`-D gssapi=disabled`: libsoup3 defaults to building with GSSAPI support, which requires Kerberos (as does the test suite).

`-D sysprof=disabled`: libsoup3 will automatically download a git version of [sysprof](#) if git is available or use the installed version if it is present on the system. If you need profiling, remove this option.

Contents

Installed Programs: None

Installed Libraries: libsoup-3.0.so

Installed Directories: /usr/include/libsoup-3.0, /usr/share/doc/libsoup-3.4.4 (only installed if gi-docgen is available)

Short Descriptions

libtirpc-1.3.5

Introduction to libtirpc

The libtirpc package contains libraries that support programs that use the Remote Procedure Call (RPC) API. It replaces the RPC, but not the NIS library entries that used to be in glibc.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/libtirpc/libtirpc-1.3.5.tar.bz2>
- Download MD5 sum: 59a5aba60d99621963d0109f95b622f2
- Download size: 552 KB
- Estimated disk space required: 7.4 MB
- Estimated build time: less than 0.1 SBU

libtirpc Dependencies

Optional

[MIT Kerberos V5-1.21.3](#) for the GSSAPI

Installation of libtirpc

Note

If updating this package, you will also need to update any existing version of [rpcbind-1.2.7](#)

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --disable-static \
            --disable-gssapi \
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-gssapi`: This switch is needed if no GSSAPI is installed. Remove this switch if you have one installed (for example [MIT Kerberos V5-1.21.3](#)) and you wish to use it.

Contents

Installed Programs: None

Installed Libraries: libtirpc.so

Installed Directory: /usr/include/tirpc

Short Descriptions

Introduction to neon

neon is an HTTP and WebDAV client library, with a C interface.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://notroj.github.io/neon/neon-0.33.0.tar.gz>
- Download MD5 sum: f0c6f84835f93892c22750ff60955898
- Download size: 892 KB
- Estimated disk space required: 8.6 MB (additional 53 MB for tests)
- Estimated build time: 0.1 SBU (add 0.4 SBU for tests)

neon Dependencies

Optional

[GnuTLS-3.8.7.1](#), [libxml2-2.13.3](#), [MIT Kerberos V5-1.21.3](#), [nss-3.103](#) (for some tests), [xmlto-0.0.29](#) (to regenerate the documentation), [libproxy](#), and [PaKChoiS](#)

Installation of neon

Install neon by running the following commands:

```
./configure --prefix=/usr \
            \
            \
            \
            \
            \
            make
```

To test the results, issue: `make check`.

If you wish to regenerate the documentation, issue:

```
make docs
```

Now, as the `root` user:

```
make install
```

Command Explanations

`--with-ssl`: This switch enables SSL support using OpenSSL. GnuTLS can be used instead, by passing `--with-ssl=gnutls` and `--with-ca-bundle=/etc/pki/tls/certs/ca-bundle.crt` to the `configure` script.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: `neon-config`

Installed Library: `libneon.so`

Installed Directories: `/usr/include/neon` and `/usr/share/doc/neon-0.33.0`

Short Descriptions

<code>neon-config</code>	is a script that provides information about an installed copy of the neon library
<code>libneon.so</code>	is used as a high-level interface to common HTTP and WebDAV methods

nghttp2-1.62.1

Introduction to nghttp2

nghttp2 is an implementation of HTTP/2 and its header compression algorithm, HPACK.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/nghttp2/nghttp2/releases/download/v1.62.1/nghttp2-1.62.1.tar.xz>
- Download MD5 sum: 965f9a9e0f0eef12d8976fb33e9312ef
- Download size: 1.5 MB
- Estimated disk space required: 19 MB
- Estimated build time: less than 0.1 SBU

nghttp2 Dependencies

Recommended

[libxml2-2.13.3](#)

Optional

The following are only used if building the full package instead of only the main libraries: [Boost-1.86.0](#), [c-ares-1.33.0](#), [cython-3.0.11](#), [jansson-2.14](#), [libevent-2.1.12](#), [sphinx-8.0.2](#), [jemalloc](#), [libev](#), [mruby](#), and [Spdylay](#).

Installation of nghttp2

Install nghttp2 by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --enable-lib-only \
            --docdir=/usr/share/doc/nghttp2-1.62.1 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-lib-only`: only build libnghttp2. Omit this switch if you'd like to build the example applications, Python bindings, or the C++ asio library.

Contents

Installed Programs: None

Installed Libraries: libnghttp2.so

Installed Directories: /usr/include/nghttp2, /usr/share/nghttp2, and /usr/share/doc/nghttp2-1.62.1

Short Descriptions

`libnghttp2.so` an implementation of the Hypertext Transfer Protocol version 2 in C

rpcsvc-proto-1.4.4

Introduction to rpcsvc-proto

The rpcsvc-proto package contains the rpcsvc protocol files and headers, formerly included with glibc, that are not included in replacement [libtirpc-1.3.5](#), along with the rpcgen program.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/thkukuk/rpcsvc-proto/releases/download/v1.4.4/rpcsvc-proto-1.4.4.tar.xz>
- Download MD5 sum: bf908de360308d909e9cc469402ff2ef
- Download size: 168 KB
- Estimated disk space required: 2.2 MB
- Estimated build time: less than 0.1 SBU

Installation of rpcsvc-proto

Install rpcsvc-proto by running the following commands:

```
./configure --sysconfdir=/etc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: rpcgen

Installed Libraries: None

Installed Directories: /usr/include/rpcsvc

Short Descriptions

`rpcgen` Generates C code to implement the RPC protocol

Serf-1.3.10

Introduction to Serf

The Serf package contains a C-based HTTP client library built upon the Apache Portable Runtime (APR) library. It multiplexes connections, running the read/write communication asynchronously. Memory copies and transformations are kept to a minimum to provide high performance operation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist-serf-serf-1.3.10.tar.bz2>
- Download MD5 sum: 5320087299084c297eff8e1dacfab1af
- Download size: 148 KB
- Estimated disk space required: 2.4 MB
- Estimated build time: less than 0.1 SBU

Serf Dependencies

Required

[Apr-Util-1.6.3](#) and [SCons-4.8.0](#)

Optional

[MIT Kerberos V5-1.21.3](#) (for GSSAPI support)

Installation of Serf

Install Serf by running the following commands:

```
sed -i "/Append/s:RPATH=libdir,::" SConstruct &&
sed -i "/Default/s:lib_static,::" SConstruct &&
sed -i "/Alias/s:install_static,::" SConstruct &&

scons PREFIX=/usr
```

This package does not come with a functional test suite.

Now, as the `root` user:

```
scons PREFIX=/usr install
```

Command Explanations

`sed -i "....":` The first command removes the runtime path from a shared library and the next two commands disable building and installing of the static library.

`GSSAPI=/usr`: Use this switch if you have installed a GSSAPI library and you want serf to use it.

Contents

Installed Programs: None

Installed Library: libserf-1.so

Installed Directory: /usr/include/serf-1

Short Descriptions

`libserf-1.so` contains the Serf API functions

uhttpmock-0.11.0

Introduction to uhttpmock

The uhttpmock package contains a library for mocking web service APIs which use HTTP or HTTPS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://tecnocode.co.uk/downloads/uhttpmock/uhttpmock-0.11.0.tar.xz>
- Download MD5 sum: 214226c73bbe9cd62ba987ce717dfc1e
- Download size: 48 KB
- Estimated disk space required: 1.1 MB
- Estimated build time: less than 0.1 SBU (with tests)

uhttpmock Dependencies

Required

[libsoup-3.4.4](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of uhttpmock

Install uhttpmock by running the following commands:

```

mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gtk_doc=false &&
ninja

```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Libraries: libuhttpmock-1.0.so

Installed Directories: /usr/include/libuhttpmock-1.0 and /usr/share/gtk-doc/html/libuhttpmock-1.0

Short Descriptions

`libuhttpmock-1.0.so` contains the uhttpmock API functions

Chapter 18. Text Web Browsers

People who are new to Unix-based systems tend to ask the question "Why on earth would I want a text-mode browser? I'm going to compile X and use Firefox/Falkon/Whatever!". Those who have been around systems for a while know that when (not if) you manage to mess up your graphical browser install and you need to look up some information on the web, a console based browser will save you. Also, there are quite a few people who prefer to use one of these browsers as their primary method of browsing; either to avoid the clutter and bandwidth which accompanies images or because they may use a text-to-speech synthesizer which can read the page to them (of use for instance to partially sighted or blind users). In this chapter you will find installation instructions for two console web browsers:

Links-2.30

Introduction to Links

Links is a text and graphics mode WWW browser. It includes support for rendering tables and frames, features background downloads, can display colors and has many other features.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://links.twibright.com/download/links-2.30.tar.bz2>
- Download MD5 sum: dc56041551980c74dd354cd7c2882539
- Download size: 6.3 MB
- Estimated disk space required: 36 MB
- Estimated build time: 0.2 SBU

Links Dependencies

Recommended

[libevent-2.1.12](#)

Optional

Graphics mode requires at least one of [GPM-1.20.7](#) (mouse support to be used with a framebuffer-based console), [SVGAlib](#), [DirectFB](#), and [a graphical environment](#)

For decoding various image formats Links can utilize [libavif-1.1.1](#), [libpng-1.6.43](#), [libjpeg-turbo-3.0.1](#), [librsvg-2.58.3](#), and [libtiff-4.6.0](#)

For decompressing web pages that are compressed with Brotli, Links can utilize [Brotli-1.1.0](#)

Installation of Links

Install Links by running the following commands:

```
./configure --prefix=/usr --mandir=/usr/share/man &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -v -d -m755 /usr/share/doc/links-2.30 &&
install -v -m644 doc/links_cal/* KEYS BRAILLE_HOWTO \
/usr/share/doc/links-2.30
```

Command Explanations

`--enable-graphics`: This switch enables support for graphics mode.

Configuring Links

Config Files

```
~/.links/*
```

Configuration Information

Links stores its configuration in per-user files in the `~/.links` directory. These files are created automatically when `links` is run for the first time.

Contents

Installed Program: `links`

Installed Libraries: None

Installed Directories: `/usr/share/doc/links-2.30`

Short Descriptions

`links` is a text and graphics mode WWW browser

Lynx-2.9.2

Introduction to Lynx

Lynx is a text based web browser.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://invisible-mirror.net/archives/lynx/tarballs/lynx2.9.2.tar.bz2>
- Download MD5 sum: 3ce01505e82626ca4d7291d7e649c4c9
- Download size: 2.7 MB
- Estimated disk space required: 25 MB
- Estimated build time: 0.4 SBU

Optional

[GnuTLS-3.8.7.1](#) (experimental, to replace openssl), [Zip-3.0](#), [UnZip-6.0](#), an [MTA](#) (that provides a `sendmail` command), and [Sharutils-4.15.2](#) (for the `uudecode` program)

Installation of Lynx

Install Lynx by running the following commands:

```
./configure --prefix=/usr          \
--sysconfdir=/etc/lynx  \
--with-zlib          \
--with-bzlib          \
--with-ssl           \
--with-screen=ncursesw \
--enable-locale-charset \
--datadir=/usr/share/doc/lynx-2.9.2 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install-full
chgrp -v -R root /usr/share/doc/lynx-2.9.2/lynx_doc
```

Command Explanations

`--sysconfdir=/etc/lynx`: This parameter is used so that the configuration files are located in `/etc/lynx` instead of `/usr/etc`.

`--datadir=/usr/share/doc/lynx-2.9.2`: This parameter is used so that the documentation files are installed into `/usr/share/doc/lynx-2.9.2` instead of `/usr/share/lynx_{doc,help}`.

`--with-zlib`: This enables support for linking `libz` into Lynx.

`--with-bzlib`: This enables support for linking `libbz2` into Lynx.

`--with-ssl`: This enables support for linking SSL into Lynx.

`--with-screen=ncursesw`: This switch enables the use of advanced wide-character support present in the system NCurses library. This is needed for proper display of characters and line wrapping in multibyte locales.

`--enable-locale-charset`: This switch allows Lynx to deduce the proper character encoding for terminal output from the current locale. A configuration step is still needed (see below), but unlike the situation without this switch, the configuration step becomes the same for all users (without the switch one must specify the display character set explicitly). This is important for environments such as a LiveCD, where the amount of system-specific configuration steps has to be reduced to the minimum.

`--enable-ipv6`: This switch allows Lynx to use IPv6, along with IPv4. Use it if your ISP provides an IPv6 configuration.

`--enable-nls`: This switch allows Lynx to print translated messages (such as questions about cookies and SSL certificates).

`--with-gnutls`: This enables experimental support for linking GnuTLS into Lynx. Remove the `--with-ssl` switch if you want to use gnutls.

`make install-full`: In addition to the standard installation, this target installs the documentation and help files.

`chgrp -v -R root /usr/share/doc/lynx-2.9.2/lynx_doc` : This command corrects the improper group ownership of installed documentation files.

Configuring Lynx

Config Files

`/etc/lynx/lynx.cfg`

Configuration Information

The proper way to get the display character set is to examine the current locale. However, Lynx does not do this by default. As the `root` user, change this setting:

```
sed -e '/#LOCALE/      a LOCALE_CHARSET:TRUE' \
-i /etc/lynx/lynx.cfg
```

The built-in editor in Lynx [Breaks Multibyte Characters](#). This issue manifests itself in multibyte locales, e.g., as the Backspace key not erasing non-ASCII characters properly, and as incorrect data being sent to the network when one edits the contents of text areas. The only solution to this problem is to configure Lynx to use an external editor (bound to the "Ctrl+X e" key combination by default). Still as the `root` user:

```
sed -e '/#DEFAULT_ED/ a DEFAULT_EDITOR:vi' \
-i /etc/lynx/lynx.cfg
```

Lynx handles the following values of the `DEFAULT_EDITOR` option specially by adding cursor-positioning arguments: `emacs`, `jed`, `jmacs`, `joe`, `jove`, `jpico`, `jstar`, `nano`, `pico`, `rjoe`, `vi` (but not `vim`: in order to position the cursor in [Vim-9.1.0660](#), set this option to `vi`).

By default, Lynx doesn't save cookies between sessions. Again as the `root` user, change this setting:

```
sed -e '/#PERSIST/   a PERSISTENT_COOKIES:TRUE' \
-i /etc/lynx/lynx.cfg
```

Many other system-wide settings such as proxies can also be set in the `/etc/lynx/lynx.cfg` file.

Contents

Installed Program: lynx

Installed Libraries: None

Installed Directories: /etc/lynx and /usr/share/doc/lynx-2.9.2

Short Descriptions

`lynx` is a general purpose, text-based, distributed information browser for the World Wide Web

Chapter 19. Mail/News Clients

Mail Clients help you retrieve (Fetchmail), sort (Procmail), read and compose responses (Heirloom mailx, Mutt, Pine, Kmail, Balsa, Evolution, SeaMonkey) to email.

News clients also help you retrieve, sort, read and compose responses, but these messages travel through USENET (a worldwide bulletin board system) using the Network News Transfer Protocol (NNTP).

Fetchmail-6.4.39

Introduction to Fetchmail

The Fetchmail package contains a mail retrieval program. It retrieves mail from remote mail servers and forwards it to the local (client) machine's delivery system, so it can then be read by normal mail user agents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/fetchmail/fetchmail-6.4.39.tar.xz>
- Download MD5 sum: 33a916a14b91ba932b970fe02d28f058
- Download size: 1.2 MB
- Estimated disk space required: 12 MB including tests
- Estimated build time: 0.1 SBU including tests

Fetchmail Dependencies

Recommended

a local MDA ([Procmail-3.24](#))

Optional

[MIT Kerberos V5-1.21.3](#) and [libgssapi](#)

Optional (for running fetchmailconf)

[Python-3.12.5](#), built after [Tk-8.6.14](#), with the [py-future](#) package

Installation of Fetchmail

Create a dedicated user for the fetchmail program. Issue the following commands as the `root` user:

```
useradd -c "Fetchmail User" -d /dev/null -g nogroup \
-s /bin/false -u 38 fetchmail
```

Install Fetchmail by running the following commands:

```
PYTHON=python3 \
./configure --prefix=/usr \
--enable-fallback=procmail &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
chown -v fetchmail:nogroup /usr/bin/fetchmail
```

Command Explanations

`PYTHON=python3`: a version of Python is *required*, but only used to install a module to allow `fetchmailconf` to be run. That module is unmaintained and should not be used.

`--enable-fallback=procmail`: This tells Fetchmail to hand incoming mail to Procmail for delivery, if the port 25 mail server is not present or not responding.

Configuring Fetchmail

Config Files

`~/.fetchmailrc`

Configuration Information

Note

If you are connecting to a mailserver that supports SSL/TLS-wrapped or "implicit" mode on a dedicated port (default 993) you should use `fetchmail --ssl` or add the option 'ssl' in an rfile.

```
cat > ~/.fetchmailrc << "EOF"

# The logfile needs to exist when fetchmail is invoked, otherwise it will
# dump the details to the screen. As with all logs, you will need to rotate
# or clear it from time to time.
set logfile fetchmail.log
set no bouncemail
# You probably want to set your local username as the postmaster
set postmaster <username>

poll SERVERNAME :
    user <isp_username> pass <password>;
    mda "/usr/bin/procmail -f %F -d %T";
EOF

touch ~/fetchmail.log      &&
chmod -v 0600 ~/.fetchmailrc
```

This is an example configuration that should suffice for most people. You can add as many users and servers as you need using the same syntax.

man fetchmail: Look for the section near the bottom named *CONFIGURATION EXAMPLES*. It gives some quick examples. There are countless other configuration options once you get used to it.

If you expect to receive very little mail you can invoke fetchmail when you wish to receive any incoming mail. More commonly, it is either invoked in daemon mode with the `-d` option either on the command line, or in `.fetchmailrc` (see 'DAEMON MODE' in man `fetchmailconf`), or alternatively it is invoked from a cron job.

Contents

Installed Programs: `fetchmail` and `fetchmailconf`

Installed Libraries: None

Installed Directories: None

Short Descriptions

- | | |
|----------------------------|--|
| <code>fetchmail</code> | when executed as a user, this will source <code>~/.fetchmailrc</code> and download the appropriate mail |
| <code>fetchmailconf</code> | is intended to assist you in setting up and editing a <code>~/.fetchmailrc</code> configuration file, by using a Tk GUI interface. It is written for Python and the Tkinter module but is labelled with a WARNING that it needs to be updated for fetchmail 6.4's SSL options and other recent new options |

mailx-12.5

Introduction to Heirloom mailx

The Heirloom mailx package (formerly known as the Nail package) contains `mailx`, a command-line Mail User Agent derived from Berkeley Mail. It is intended to provide the functionality of the POSIX `mailx` command with additional support for MIME messages, IMAP (including caching), POP3, SMTP, S/MIME, message threading/sorting, scoring, and filtering. Heirloom mailx is especially useful for writing scripts and batch processing.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://anduin.linuxfromscratch.org/BLFS/mailx/heirloom-mailx_12.5.orig.tar.gz
- Download MD5 sum: 29a6033ef1412824d02eb9d9213cb1f2
- Download size: 317 KB
- Estimated disk space required: 3.2 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/heirloom-mailx-12.5-fixes-1.patch>

Heirloom mailx Dependencies

Optional

[nss-3.103](#), [MIT Kerberos V5-1.21.3](#) (for IMAP GSSAPI authentication), and an [MTA](#)

Installation of Heirloom mailx

Note

This package does not support parallel build.

Install Heirloom mailx by running the following commands.

```
patch -Np1 -i ../heirloom-mailx-12.5-fixes-1.patch &&
sed 's@<openssl1@<openssl-1.0/openssl@' \
-i openssl.c fio.c makeconfig &&
```

```
make -j1 LDFLAGS+=" -L /usr/lib/openssl/" \
      SENDMAIL=/usr/sbin/sendmail
```

This package does not come with a test suite.

Now, as the `root` user:

```
make PREFIX=/usr UCBINSTALL=/usr/bin/install install &&
ln -v -sf mailx /usr/bin/mail &&
ln -v -sf mailx /usr/bin/nail &&
install -v -m755 -d    /usr/share/doc/heirloom-mailx-12.5 &&
install -v -m644 README /usr/share/doc/heirloom-mailx-12.5
```

Command Explanations

`make SENDMAIL=/usr/sbin/sendmail`: This changes the default MTA path of `/usr/lib/sendmail`.

`make PREFIX=/usr UCBINSTALL=/usr/bin/install install`: This changes the default installation path of `/usr/local` and the default `install` command path of `/usr/ucb`.

Configuring Heirloom mailx

Config Files

`/etc/nail.rc` and `~/.mailrc`

Configuration Information

For displaying mails, mailx uses a pager program. Since the default of `pg` is not available on a LFS system, its required to specify which pager is to use. By default, there is `more` and the more comfortable `less` installed. If the variable `PAGER` is not set in `/etc/profile` or `~/.bash_profile`, or if it should be another pager just for reading mails, it can be set system wide in `/etc/nail.rc`:

```
echo "set PAGER=<more|less>" >> /etc/nail.rc
```

or individually for the actual user in `~/.mailrc`:

```
echo "set PAGER=<more|less>" >> ~/.mailrc
```

Other interesting options to set in the config files might be `EDITOR` and `MAILDIR`.

If not set in the environment for other packages, the default editor can be set by:

```
echo "set EDITOR=<vim|nano|...>" >> /etc/nail.rc
```

Depending on which kind of `MTA` is installed, it might be required to set the `MAILDIR` variable so mailx is able to find the mails:

```
echo "set MAILDIR=Maildir" >> /etc/nail.rc
```

Contents

Installed Programs: mail, mailx and nail

Installed Libraries: None

Installed Directories: None

Short Descriptions

- `mailx` is a command-line mail user agent compatible with the `mailx` command found on commercial Unix versions
- `mail` is a symbolic link to `mailx`
- `nail` is a symbolic link to `mailx`

Mutt-2.2.13

Introduction to Mutt

The Mutt package contains a Mail User Agent. This is useful for reading, writing, replying to, saving, and deleting your email.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://bitbucket.org/mutt/mutt/downloads/mutt-2.2.13.tar.gz>
 - Download MD5 sum: 0d33c808d5556c283c01542c7730ccbd
 - Download size: 5.3 MB
 - Estimated disk space required: 48 MB
 - Estimated build time: 0.2 SBU

Mutt Dependencies

Recommended (for a text version of the manual)

[Lynx-2.9.2](#), or [Links-2.30](#) (or [W3m](#), or [ELinks](#)) - please read the Note.

Optional

[Aspell-0.60.8.1](#), [Cyrus SASL-2.1.28](#), [DocBook-utils-0.6.14](#), [GDB-15.1](#), [GnuPG-2.4.5](#), [GnuTLS-3.8.7.1](#), [GPGME-1.23.2](#), [libidn-1.42](#), [MIT Kerberos V5-1.21.3](#), an [MTA](#) (that provides a `sendmail` command), [slang-2.3.3](#), [SQLite-3.46.1](#), [libgssapi](#), [Mixmaster](#), [ODBM](#) or [Tokyo Cabinet](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/mutt>

Installation of Mutt

Note

Mutt ships with an HTML version of its manual, but the text version is no longer provided because differences in the formatting from different text browsers compared to the shipped version caused complaints. To get a text file, the following are used in order of preference: lynx with overstriking (backspaces) for emphasis, w3m or elinks: the latter two apparently provide plain text. Plain text is generally preferred unless reading the HTML manual, so the instructions below use lynx if available, or else links to produce plain text.

Mutt requires a group named `mail`. You can add this group, if it does not exist, with this command (as the `root` user):

```
groupadd -q 34 mail
```

If you did not install an [MTA](#), you need to modify the ownership of `/var/mail` with this command:

```
chgrp -v mail /var/mail
```

Install Mutt by running the following commands:

To ensure that a plain text manual is created when using lynx or to otherwise use links to produce it (instead of elinks), run the following command:

```
sed -e 's/ -with_backspaces//' \
-e 's/elinks/links/' \
-e 's/-no-numbering -no-references//' \
-i doc/Makefile.in
```

Now configure and build the application:

```
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --with-docdir=/usr/share/doc/mutt-2.2.13 \
            --with-ssl
```

```
--enable-external-dotlock      \
--enable-pop                  \
--enable-imap                  \
--enable-hcache                \
--enable-sidebar              &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Note

If you used a DESTDIR method to only install to a temporary location as a regular user (as part of a package management process), you will need to run the following as the `root` user after completing the real install:

```
chown root:mail /usr/bin/mutt_dotlock &&
chmod -v 2755 /usr/bin/mutt_dotlock
```

An info file is now installed, so you will also need to recreate the `/usr/share/info/dir` as described when Texinfo was installed in LFS.

Command Explanations

`sed ... -e 's/_with_backspaces//'` ...: This turns off the backspaces used for overstriking when lynx is used, resulting in readable plain text when using `view`.

`sed ... -e 's/_elinks/_links/'` ...: This allows `links` to be run instead of `elinks` which is not in the book.

`sed ... -e 's/_no-numbering _no-references//'` ...: This removes switches which are not understood by `links`.

`--enable-external-dotlock`: In some circumstances the mutt-dotlock program is not created. This switch ensures it is always created.

`--enable-pop`: This switch enables POP3 support.

`--enable-imap`: This switch enables IMAP support.

`--enable-hcache`: This switch enables header caching.

`--enable-sidebar`: This switch enables support for the sidebar (a list of mailboxes). It is off by default, but can be turned on by `:set sidebar_visible` in mutt (and off again with '`:unset`'), or it can be enabled in `~/.muttrc`.

`--with-ssl`: This parameter adds SSL/TLS support from openssl in POP3/IMAP/SMTP.

`--enable-autocrypt --with-sqlite3`: These two switches add support for passive protection against data collection, using gnupg and gpgme (gpgme is enabled by autocrypt). See [The Manual](#).

`--enable-gpgme`: This switch enables GPG support through the GPGME package. Use this switch if you want GPG support in Mutt.

`--enable-smtp`: This switch enables SMTP relay support.

`--with-idn2`: Use this parameter if both libidn and libidn2 have been installed, and you wish to use libidn2 here.

`--with-sasl`: This parameter adds authentication support from [Cyrus SASL-2.1.28](#) in POP3/IMAP/SMTP if they are enabled. Depending on the server configuration, this may not be needed for POP3 or IMAP. However, it is needed for SMTP authentication.

Configuring Mutt

Config Files

`/etc/Muttrc`, `~/.muttrc`, `/etc/mime.types`, and `~/.mime.types`

Configuration Information

No changes in these files are necessary to begin using Mutt. When you are ready to make changes, the man page for `muttrc` is a good starting place.

In order to utilize GnuPG, use the following command:

```
cat /usr/share/doc/mutt-2.2.13/samples/gpg.rc >> ~/.muttrc
```

Contents

Installed Programs: `flea`, `mutt`, `mutt_dotlock`, `muttbug`, `pgpwrap`, `mutt_pgpring`, and `smime_keys`

Installed Libraries: None

Installed Directories: /usr/share/doc/mutt-2.2.13

Short Descriptions

<code>flea</code>	is a script showing where to report bugs
<code>mutt</code>	is a Mail User Agent (MUA) which enables you to read, write and delete your email
<code>mutt_dotlock</code>	implements the mail spool file lock
<code>muttbug</code>	is a script identical to <code>flea</code>
<code>pgpwrap</code>	prepares a command line for the GnuPG-2.4.5 utilities
<code>pgpring</code>	is a key ring dumper for PGP . It is not needed for GnuPG-2.4.5
<code>smime_keys</code>	manages a keystore for S/MIME certificates

Procmail-3.24

Introduction to Procmail

The Procmail package contains an autonomous mail processor. This is useful for filtering and sorting incoming mail.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/BuGlessRB/procmail/archive/refs/tags/v3.24/procmail-3.24.tar.gz>
- Download MD5 sum: e38b8739e5c6400e3586c5fd9810c1e0
- Download size: 296 KB
- Estimated disk space required: 2.1 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/procmail-3.24-consolidated_fixes-1.patch

Procmail Dependencies

Recommended

A [MTA](#) that installs sendmail

Optional

[libnsl-2.0.1](#)

Installation of Procmail

This package does not come with a test suite.

Install Procmail by running the following commands as the `root` user:

```
patch -Np1 -i ../procmail-3.24-consolidated_fixes-1.patch &&
```

```
make LOCKINGTEST=/tmp MANDIR=/usr/share/man install      &&
make install-suid
```

Command Explanations

`make LOCKINGTEST=/tmp install`: This prevents `make` from asking you where to test file-locking patterns.

`make install-suid`: Modifies permissions of the installed files.

Configuring Procmail

Config Files

`/etc/procmailrc` and `~/.procmailrc`

Configuration Information

Recipes have to be written and placed in `~/.procmailrc` for execution. The `procmailex` man page is the starting place to learn how to write recipes. For additional information, see also <https://pm-doc.sourceforge.net/>.

Contents

Installed Programs: `formail`, `lockfile`, `mailstat` and `procmail`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>formail</code>	is a filter that can be used to format mail into mailbox format
<code>lockfile</code>	is a utility that can lock a file for single use interactively or in a script
<code>mailstat</code>	prints a summary report of mail that has been filtered by <code>procmail</code> since the last time <code>mailstat</code> was run
<code>procmail</code>	is an autonomous mail processor. It performs all the functions of an MDA (Mail Delivery Agent)

Other Mail and News Programs

[Balsa-2.6.4](#) is a GTK2 based mail client.

[seamonkey-2.53.18.2](#) includes both a mail client and newsreader in its installation.

[Thunderbird-128.1.0esr](#) is a mail/news client based on the Mozilla code base.

[Evolution-3.52.4](#) is a mail client and organizer from the GNOME desktop.

Part V. Servers

Chapter 20. Major Servers

Major servers are the programs that provide content or services to users or other programs.

Apache-2.4.62

Introduction to Apache HTTPD

The Apache HTTPD package contains an open-source HTTP server. It is useful for creating local intranet web sites or running huge web serving operations.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist/httpd/httpd-2.4.62.tar.bz2>
- Download MD5 sum: cded7afa23c13c4854008d95a69ce016

- Download size: 7.2 MB
- Estimated disk space required: 85 MB
- Estimated build time: 0.4 SBU (Using parallelism=4)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/httpd-2.4.62-blfs_layout-1.patch

Apache HTTPD Dependencies

Required

[Apr-Util-1.6.3](#) and [pcre2-10.44](#)

Optional

[Brotli-1.1.0](#), [Doxygen-1.12.0](#), [jansson-2.14](#), [libxml2-2.13.3](#), [Lua-5.4.7](#), [Lynx-2.9.2](#) or [Links-2.30](#) or [ELinks](#), [nghttp2-1.62.1](#), [OpenLDAP-2.6.8](#) ([Apr-Util-1.6.3](#) needs to be installed with ldap support), [rsync-3.3.0](#), [Berkeley DB](#) (deprecated), and [Distcache](#)

Installation of Apache HTTPD

For security reasons, running the server as an unprivileged user and group is strongly encouraged. Create the following group and user using the following commands as `root`:

```
groupadd -g 25 apache &&
useradd -c "Apache Server" -d /srv/www -g apache \
-s /bin/false -u 25 apache
```

Build and install Apache HTTPD by running the following commands:

```
patch -Np1 -i ../../httpd-2.4.62-blfs_layout-1.patch &&
sed '/dir.*CFG_PREFIX/s:@#@' -i support/apxs.in &&
sed -e '/HTTPD_ROOT/s:${ap_prefix}:etc/httpd:' \
-e '/SERVER_CONFIG_FILE/s:${rel_sysconfdir}/::' \
-e '/AP_TYPES_CONFIG_FILE/s:${rel_sysconfdir}/::' \
-i configure &&
sed -e '/encoding.h/a # include <libxml/xmlstring.h>' \
-i modules/filters/mod_xml2enc.c &&
./configure --enable-authnz-fcgi \
--enable-layout=BLFS \
--enable-mods-shared="all cgi" \
--enable-mpms-shared=all \
--enable-suexec=shared \
--with-apr=/usr/bin/apr-1-config \
--with-apr-util=/usr/bin/apu-1-config \
--with-suexec-bin=/usr/lib/httpd/suexec \
--with-suexec-caller=apache \
--with-suexec-docroot=/srv/www \
--with-suexec-logfile=/var/log/httpd/suexec.log \
--with-suexec-uidmin=100 \
--with-suexec-userdir=public_html &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
mv -v /usr/sbin/suexec /usr/lib/httpd/suexec &&
chgrp apache /usr/lib/httpd/suexec &&
chmod 4754 /usr/lib/httpd/suexec &&
chown -v -R apache:apache /srv/www
```

Command Explanations

`sed '/dir.*CFG_PREFIX/s@^@#@' ...`: Forces the apxs utility to use absolute pathnames for modules, when instructed to do so.

`sed -e '/HTTPD_ROOT/s ...'`: Fixes some paths.

`sed -e '/encoding.h/a ...'; Fix building against libxml-2.12.x.`

`--enable-authnz-fcgi`: Build FastCGI authorizer-based authentication and authorization (mod_authnz_fcgi.so fast CGI module).

`--enable-mods-shared="all cgi"`: The modules should be compiled and used as Dynamic Shared Objects (DSOs) so they can be included and excluded from the server using the run-time configuration directives.

`--enable-mpms-shared=all`: This switch ensures that all MPM (Multi Processing Modules) are built as Dynamic Shared Objects (DSOs), so the user can choose which one to use at runtime.

`--enable-suexec`: This switch enables building of the Apache suEXEC module which can be used to allow users to run CGI and SSI scripts under user IDs different from the user ID of the calling web server.

`--with-suexec-*`: These switches control suEXEC module behavior, such as default document root, minimal UID that can be used to run the script under the suEXEC. Please note that with minimal UID 100, you can't run CGI or SSI scripts under suEXEC as the `apache` user.

`... /usr/lib/httpd/suexec`: These commands put `suexec` wrapper into proper location, since it is not meant to be run directly. They also adjust proper permissions of the binary, making it setgid `apache`.

`chown -R apache:apache /srv/www`: By default, the installation process installs files (documentation, error messages, default icons, etc.) with the ownership of the user that extracted the files from the tar file. If you want to change the ownership to another user, you should do so at this point. The only requirement is that the document directories need to be accessible by the `httpd` process with (r-x) permissions and files need to be readable (r--) by the `apache` user.

Configuring Apache

Config Files

`/etc/httpd/httpd.conf` and `/etc/httpd/extr/*`

Configuration Information

See <file:///usr/share/httpd/manual/configuring.html> for detailed instructions on customising your Apache HTTP server configuration file.

There is no reason, at least for internet facing sites, not to use SSL encryption. Setting up a secured website does not cost anything except installing one additional small tool and a few minutes of configuration work. Use this guideline at https://wiki.linuxfromscratch.org/blfs/wiki/Securing_a_website to create world-wide accepted certificates and renew them on a regular basis.

Systemd Unit

If you want the Apache server to start automatically when the system is booted, install the `httpd.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-httpd
```

Contents

Installed Programs:	ab, apachectl, apxs, checkgid, dbmmanage, fcgistarter, htcacheclen, httdbm, httdigest, htpasswd, httpd, httxtdb, logresolve, and rotatelogs
Installed Libraries:	Several libraries under /usr/lib/httpd/modules/
Installed Directories:	/etc/httpd, /srv/www, /usr/include/httpd, /usr/lib/httpd, /usr/share/httpd, /var/log/httpd, and /var/run/httpd

Short Descriptions

<code>ab</code>	is a tool for benchmarking your Apache HTTP server
<code>apachectl</code>	is a front end to the Apache HTTP server which is designed to help the administrator control the functioning of the Apache httpd daemon
<code>apxs</code>	is a tool for building and installing extension modules for the Apache HTTP server

<code>checkgid</code>	is a program that checks whether it can setgid to the group specified. This is to see if it is a valid group for Apache2 to use at runtime. If the user (should be run as superuser) is in that group, or can setgid to it, it will return 0
<code>dbmmanage</code>	is used to create and update the DBM format files used to store usernames and passwords for basic authentication of HTTP users
<code>fcgistarterm</code>	is a tool to start a FastCGI program
<code>htcacheclean</code>	is used to clean up the disk cache
<code>htdbm</code>	is used to manipulate the DBM password databases
<code>htdigest</code>	is used to create and update the flat-files used to store usernames, realms and passwords for digest authentication of HTTP users
<code>htpasswd</code>	is used to create and update the flat-files used to store usernames and passwords for basic authentication of HTTP users
<code>httpd</code>	is the Apache HTTP server program
<code>httxt2dbm</code>	is used to generate DBM files from text, for use in RewriteMap
<code>logresolve</code>	is a post-processing program to resolve IP-addresses in Apache's access log files
<code>rotatelogs</code>	is a simple program for use in conjunction with Apache's piped log file feature
<code>susexec</code>	allows users to run CGI and SSI applications as a different user

BIND-9.20.0

Introduction to BIND

The BIND package provides a DNS server and client utilities. If you are only interested in the utilities, refer to the [BIND Utilities-9.20.0](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.isc.org/isc/bind9/9.20.0/bind-9.20.0.tar.xz>
- Download MD5 sum: 2c8d94d1524cbbae4d76cd74955bb6d9
- Download size: 5.5 MB
- Estimated disk space required: 138 MB (22 MB installed)
- Estimated build time: 0.5 SBU (with parallelism=4; about 40 minutes somewhat processor independent, to run the complete test suite)

BIND Dependencies

Required

[libcurl-0.14.0](#) and [libuv-1.48.0](#)

Recommended

[JSON-C-0.17](#), [libcap-2.70 with PAM](#), and [nghttp2-1.62.1](#)

Optional

[cURL-8.9.1](#), [libidn2-2.3.7](#), [libxml2-2.13.3](#), [lmdb-0.9.31](#), [MIT Kerberos V5-1.21.3](#), [pytest-8.3.2](#), [sphinx-8.0.2](#) (required to build documentation), [cmocka](#), [geoip](#), [jemalloc](#), [w3m](#)

Optional (to run the test suite)

[Net-DNS-1.46](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/bind>

Installation of BIND

Note

Starting with bind-9.18.20, the IP addresses for B.ROOT-SERVERS.NET have changed.

Install BIND by running the following commands:

```
./configure --prefix=/usr      \
--sysconfdir=/etc      \
--localstatedir=/var      \
--mandir=/usr/share/man \
--disable-static      &&
make
```

Issue the following commands to run the complete suite of tests. First, as the `root` user, set up some test interfaces:

Note

If IPv6 is not enabled in the kernel, there will be several error messages: "RTNETLINK answers: Operation not permitted". These messages do not affect the tests.

```
bin/tests/system/ifconfig.sh up
```

The test suite may indicate some skipped tests depending on what configuration options are used. Some tests are marked "UNTESTED" or do even fail if [Net-DNS-1.46](#) is not installed. One test, `dnssec`, is known to fail. The tests require the [hypothesis](#) python module, which is not in BLFS. To work around this, the instructions below create a Python virtual environment and then install the module in there before running the tests. To run the tests, as an unprivileged user, execute:

```
python3 -m venv --system-site-packages testenv &&
source testenv/bin/activate
pip3 install hypothesis
make -k check
deactivate
```

Again as `root`, clean up the test interfaces:

```
bin/tests/system/ifconfig.sh down
```

Finally, install the package as the `root` user:

```
make install
```

Command Explanations

`--sysconfdir=/etc`: This parameter forces BIND to look for configuration files in `/etc` instead of `/usr/etc`.

`--with-libidn2`: This parameter enables the IDNA2008 (Internationalized Domain Names in Applications) support.

`--enable-fetchlimit`: Use this option if you want to be able to limit the rate of recursive client queries. This may be useful on servers which receive a large number of queries.

`--disable-doh`: Use this option if you do not have [nghttp2-1.62.1](#) installed and do not need support for DNS over HTTPS.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Configuring BIND

Config files

`named.conf`, `root.hints`, `127.0.0.0`, `rndc.conf`, and `resolv.conf`

Configuration Information

BIND will be configured to run in a `chroot` jail as an unprivileged user (`named`). This configuration is more secure in that a DNS compromise can only affect a few files in the `named` user's `HOME` directory.

Create the unprivileged user and group `named`:

```
groupadd -g 20 named &&
useradd -c "BIND Owner" -g named -s /bin/false -u 20 named &&
install -d -m770 -o named -g named /srv/named
```

Set up some files, directories and devices needed by BIND:

```
mkdir -p /srv/named &&
cd      /srv/named &&
mkdir -p dev etc/named/{slave,pz} usr/lib/engines var/run/named &&
mknod /srv/named/dev/null c 1 3 &&
mknod /srv/named/dev/urandom c 1 9 &&
chmod 666 /srv/named/dev/{null,urandom} &&
cp /etc/localtime etc
```

The `rndc.conf` file contains information for controlling `named` operations with the `rndc` utility. Generate a key for use in the `named.conf` and `rndc.conf` with the `rndc-confgen` command:

```
rndc-confgen -a -b 512 -t /srv/named
```

Create the `named.conf` file from which `named` will read the location of zone files, root name servers and secure DNS keys:

Create a zone file with the following contents:

```
cat > /srv/named/etc/named/pz/127.0.0 << "EOF"
$TTL 3D
@ IN SOA ns.local.domain. hostmaster.local.domain. (
    1           ; Serial
    8H          ; Refresh
    2H          ; Retry
    4W          ; Expire
    1D)         ; Minimum TTL
    NS          ns.local.domain.
1      PTR        localhost.
EOF
```

Create the `root.hints` file with the following commands:

Note

Caution must be used to ensure there are no leading spaces in this file.

```
cat > /srv/named/etc/named/root.hints << "EOF"
.          6D IN NS   A.ROOT-SERVERS.NET.
.          6D IN NS   B.ROOT-SERVERS.NET.
.          6D IN NS   C.ROOT-SERVERS.NET.
.          6D IN NS   D.ROOT-SERVERS.NET.
.          6D IN NS   E.ROOT-SERVERS.NET.
.          6D IN NS   F.ROOT-SERVERS.NET.
.          6D IN NS   G.ROOT-SERVERS.NET.
.          6D IN NS   H.ROOT-SERVERS.NET.
.          6D IN NS   I.ROOT-SERVERS.NET.
.          6D IN NS   J.ROOT-SERVERS.NET.
.          6D IN NS   K.ROOT-SERVERS.NET.
.          6D IN NS   L.ROOT-SERVERS.NET.
.          6D IN NS   M.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 6D IN A   198.41.0.4
A.ROOT-SERVERS.NET. 6D IN AAAA 2001:503:ba3e::2:30
B.ROOT-SERVERS.NET. 6D IN A   170.247.170.2
B.ROOT-SERVERS.NET. 6D IN AAAA 2801:1b8:10::b
C.ROOT-SERVERS.NET. 6D IN A   192.33.4.12
C.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:2::c
D.ROOT-SERVERS.NET. 6D IN A   199.7.91.13
D.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:2d::d
E.ROOT-SERVERS.NET. 6D IN A   192.203.230.10
E.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:a8::e
F.ROOT-SERVERS.NET. 6D IN A   192.5.5.241
F.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:2f::f
G.ROOT-SERVERS.NET. 6D IN A   192.112.36.4
G.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:12::d0d
H.ROOT-SERVERS.NET. 6D IN A   198.97.190.53
H.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:1::53
I.ROOT-SERVERS.NET. 6D IN A   192.36.148.17
I.ROOT-SERVERS.NET. 6D IN AAAA 2001:7fe::53
J.ROOT-SERVERS.NET. 6D IN A   192.58.128.30
J.ROOT-SERVERS.NET. 6D IN AAAA 2001:503:c27::2:30
K.ROOT-SERVERS.NET. 6D IN A   193.0.14.129
K.ROOT-SERVERS.NET. 6D IN AAAA 2001:7fd::1
L.ROOT-SERVERS.NET. 6D IN A   199.7.83.42
L.ROOT-SERVERS.NET. 6D IN AAAA 2001:500:9f::42
M.ROOT-SERVERS.NET. 6D IN A   202.12.27.33
M.ROOT-SERVERS.NET. 6D IN AAAA 2001:dc3::35
EOF
```

The `root.hints` file is a list of root name servers. This file must be updated periodically with the `dig` utility. A current copy of `root.hints` can be obtained from <https://www.internic.net/domain/named.root>. For details, consult the "BIND 9 Administrator Reference Manual".

Create or modify `resolv.conf` to use the new name server with the following commands:

Note

Replace <yourdomain.com> with your own valid domain name.

```
cp /etc/resolv.conf /etc/resolv.conf.bak &&
cat > /etc/resolv.conf << "EOF"
search <yourdomain.com>
nameserver 127.0.0.1
EOF
```

Set permissions on the `chroot` jail with the following command:

```
chown -R named:named /srv/named
```

Systemd Unit

To start the DNS server at boot, install the `named.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-named
```

Now start BIND with the following command:

```
systemctl start named
```

Testing BIND

Test out the new BIND 9 installation. First query the local host address with `dig`:

```
dig +x 127.0.0.1
```

Now try an external name lookup, taking note of the speed difference in repeated lookups due to the caching. Run the `dig` command twice on the same address:

```
dig www.linuxfromscratch.org &&
dig www.linuxfromscratch.org
```

You can see almost instantaneous results with the named caching lookups. Consult the BIND Administrator Reference Manual (see below) for further configuration options.

Administrator Reference Manual (ARM)

The ARM documentation (do not confuse with the processor architecture) is included in the source package. The documentation is in .rst format which means that it can be converted in human readable formats if [sphinx-8.0.2](#) is installed.

When BIND is set up, especially if it is going to be operating in a real live scenario, it is *highly* recommended to consult the ARM documentation. ISC provides an updated set of excellent documentation along with every release so it can be easily viewed and/or downloaded – so there is no excuse to not read the docs. The formats ISC provides are PDF, epub and html at <https://downloads.isc.org/isc/bind9/9.20.0/doc/arm/>.

Contents

Installed Programs: arpaname, ddns-confgen, delv, dig, dnssec-cds, dnssec-dsfromkey, dnssec-importkey, dnssec-keyfromlabel, dnssec-keygen, dnssec-ksr, dnssec-revoke, dnssec-settime, dnssec-signzone, dnssec-verify, host, mdig, named, named-checkconf, named-checkzone, named-compilezone, named-journalprint, named-nzd2nzf, named-rrchecker, nsec3hash, nslookup, nsupdate, rndc, rndc-confgen, and tsig-keygen (symlink)

Installed Libraries: libdns.so, libisc.so, libisccc.so, libiscfg.so, and libns.so

Installed Directories: /usr/include/{dns,dst,irs,isc,isccc,iscfg,ns}, /usr/lib/bind, and /srv/named

Short Descriptions

<code>arpaname</code>	translates IP addresses to the corresponding ARPA names
<code>ddns-confgen</code>	generates a key for use by nsupdate and named
<code>delv</code>	is a new debugging tool that is a successor to <code>dig</code>
<code>dig</code>	interrogates DNS servers
<code>dnssec-cds</code>	changes DS records for a child zone based on CDS/CDNSKEY
<code>dnssec-dsfromkey</code>	outputs the Delegation Signer (DS) resource record (RR)
<code>dnssec-importkey</code>	reads a public DNSKEY record and generates a pair of .key/.private files

<code>dnssec-keyfromlabel</code>	gets keys with the given label from a cryptography hardware device and builds key files for DNSSEC
<code>dnssec-keymgr</code>	ensures correct DNSKEY coverage based on a defined policy
<code>dnssec-ksr</code>	creates signed key response (SKR) files for offline KSK setups
<code>dnssec-revoke</code>	sets the REVOKED bit on a DNSSEC key
<code>dnssec-settime</code>	sets the key timing metadata for a DNSSEC key
<code>dnssec-signzone</code>	generates signed versions of zone files
<code>dnssec-verify</code>	verifies that a zone is fully signed for each algorithm found in the DNSKEY RRset for the zone, and that the NSEC / NSEC3 chains are complete
<code>host</code>	is a utility for DNS lookups
<code>mdig</code>	is a version of dig that allows multiple queries at once
<code>named</code>	is the name server daemon
<code>named-checkconf</code>	checks the syntax of <code>named.conf</code> files
<code>named-checkzone</code>	checks zone file validity
<code>named-compilezone</code>	is similar to <code>named-checkzone</code> , but it always dumps the zone contents to a specified file in a specified format
<code>named-journalprint</code>	prints the zone journal in human-readable form
<code>named-rrchecker</code>	reads an individual DNS resource record from standard input and checks if it is syntactically correct
<code>named-nzd2nzf</code>	converts an NZD database to NZF text format
<code>nsec3hash</code>	generates an NSEC3 hash based on a set of NSEC3 parameters
<code>nslookup</code>	is a program used to query Internet domain nameservers
<code>nsupdate</code>	is used to submit DNS update requests
<code>rndc</code>	controls the operation of BIND
<code>rndc-confgen</code>	generates <code>rndc.conf</code> files
<code>tsig-keygen</code>	is a symlink to <code>ddns-confgen</code>

Kea 2.6.1 DHCP Server

Introduction to ISC Kea DHCP Server

The ISC Kea package contains the server programs for DHCP. It is the successor of the old ISC DHCP server which is end-of-life since December 2022.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.isc.org/isc/kea/2.6.1/kea-2.6.1.tar.gz>
- Download MD5 sum: 5f4fe79ed29f5ff2802e2961d1827b25
- Download size: 9.9 MB
- Estimated disk space required: 1.5 GB (332 MB installed; add 4 GB for tests)
- Estimated build time: 4.1 SBU (with parallelism=4; add 12 SBU for tests)

Kea Dependencies

Required

[Boost-1.86.0](#) and [log4cplus-2.1.1](#)

Optional

[MIT Kerberos V5-1.21.3](#), [Valgrind-3.23.0](#); for documentation: [Doxygen-1.12.0](#), [Graphviz-12.1.0](#), and [sphinx_rtd_theme-2.0.0](#); for tests: [GoogleTest](#)

Optional database backends

[MariaDB-10.11.8](#) or [MySQL](#), and [PostgreSQL-16.4](#)

Kernel Configuration

You must have Packet Socket support. IPv6 support is optional.

```
[*] Networking support ---> [NET]
  Networking options --->
    <*/M> Packet socket [PACKET]
    [*] TCP/IP networking [INET]
    <*> The IPv6 protocol ---> [IPV6]
```

Installation of ISC Kea DHCP Server

Install ISC Kea DHCP Server by running the following commands:

```
./configure --prefix=/usr          \
            --sysconfdir=/etc    \
            --localstatedir=/var \
            --enable-shell        \
            --with-openssl       \
            --disable-static      \
            --docdir=/usr/share/doc/kea-2.6.1 &&
make
```

To test the results, you must have installed GoogleTest and kept its source. You should also have passed `--with-gtest-source=/path/to/googletest/sourcedir` to `configure` above. Run the tests with `make check`. Three tests in the TLSTest suite are known to fail.

To install the ISC Kea DHCP Server suite, issue the following commands as the `root` user:

```
make -j1 install
```

Command Explanations

`--enable-shell`: Allows building `kea-shell`, a command line interface for the control agent.

--with-openssl: Allows using OpenSSL for communicating with the control-agent and for DNS updates.

--disable-static: This switch prevents installation of static versions of the libraries.

--with-psql or --with-mysql: ISC Kea can store the leases on a database. This might be useful in large environments running a cluster of DHCP servers. Using the *memfile* backend (which is a CSV file stored locally) is possible anyhow.

`--enable-generate-docs`: If documentation is to be rebuilt, add that option. Several dependencies must be installed for generating the documentation.

make -j1 install; ISC does not recommend any form of parallel or job server options when doing the install.

Configuring ISC Kea DHCP Server

The support of IPv4, IPv6 and DDNS has been split into separate servers which runs independently from each other. Each of them has its own configuration file.

Consult the [Kea Administrator Reference Manual](#) for detailed information about the configuration of ISC Kea as it is a quite capable system. The configuration shown below is a bare minimum to get a DHCP server running but it already includes configuration for DDNS (Dynamic DNS). That setup might be working for small networks with a few clients and low traffic. For greater installations with thousands of clients, ISC Kea can be configured to use databases (mariadb or postgresql) to store the leases and build a cluster with multiple nodes. It can be integrated to [ISC Stork](#) which is a management dashboard to ISC Kea.

If you want to start the DHCP Server at boot, install the `kea-dhcpd.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-kea-dhcpd
```

Config Files

/etc/kea/kea-ctrl-agent.conf, /etc/kea/kea-dhcp4.conf, /etc/kea/kea-dhcp6.conf, and /etc/kea/kea-dhcp-ddns.conf

Kea Configuration Using Systemd Units

Four service units are used to start various daemons provided by Kea:

- Control Agent

The Control Agent is a daemon which allows the (re)configuration of the Kea DHCP service via REST API. Run `systemctl enable kea-ctrl-agent` if this daemon is needed.

- IPv4 DHCP server

This daemon handles requests for IPv4 addresses. Run `systemctl enable kea-dhcp4-server` to have it started by systemd.

- IPv6 DHCP server

This daemon handles requests for IPv6 addresses. Run `systemctl enable kea-dhcp6-server` to have it started by systemd.

- Dynamic DNS

This daemon is used to update a DNS server dynamically when Kea assigns an IP address to a device. Run `systemctl enable kea-ddns-server` to have it started by systemd.

The Netconf service is not installed because required dependencies are not covered by the current BLFS book.

Control Agent Configuration

The provided configuration could be used without changes but in BLFS, objects like sockets are stored in `/run` rather than in `/tmp`.

```
cat > /etc/kea/kea-ctrl-agent.conf << "EOF"
// Begin /etc/kea/kea-ctrl-agent.conf
{
    // This is a basic configuration for the Kea Control Agent.
    // RESTful interface to be available at http://127.0.0.1:8000/
    "Control-agent": {
        "http-host": "127.0.0.1",
        "http-port": 8000,
        "control-sockets": {
            "dhcp4": {
                "socket-type": "unix",
                "socket-name": "/run/kea4-ctrl-socket"
            },
            "dhcp6": {
                "socket-type": "unix",
                "socket-name": "/run/kea6-ctrl-socket"
            },
            "d2": {
                "socket-type": "unix",
                "socket-name": "/run/kea-ddns-ctrl-socket"
            }
        },
        "loggers": [
            {
                "name": "kea-ctrl-agent",
                "output_options": [
                    {
                        "output": "/var/log/kea-ctrl-agent.log",
                        "pattern": "%D(%Y-%m-%d %H:%M:%S.%q) %-5p %m\n"
                    }
                ],
                "severity": "INFO",
                "debuglevel": 0
            }
        ]
    }
}
// End /etc/kea/kea-ctrl-agent.conf
EOF
```

IPv4 DHCP Server Configuration

A sample configuration file is created in `/etc/kea/kea-dhcp4.conf`. Adjust the file to suit your needs or overwrite it by running the following command as the `root` user (you'll need to edit this file anyway: at least the `interfaces` field, the `ddns-qualifying-suffix` field, and almost all the fields in `Subnet4`):

```
cat > /etc/kea/kea-dhcp4.conf << "EOF"
// Begin /etc/kea/kea-dhcp4.conf
```

```
{
  "Dhcp4": {
    // Add names of your network interfaces to listen on.
    "interfaces-config": {
      "interfaces": [ "eth0", "eth2" ]
    },
    "control-socket": {
      "socket-type": "unix",
      "socket-name": "/run/kea4-ctrl-socket"
    },
    "lease-database": {
      "type": "memfile",
      "lfc-interval": 3600
    },
    "expired-leases-processing": {
      "reclaim-timer-wait-time": 10,
      "flush-reclaimed-timer-wait-time": 25,
      "hold-reclaimed-time": 3600,
      "max-reclaim-leases": 100,
      "max-reclaim-time": 250,
      "unwarned-reclaim-cycles": 5
    },
    "renew-timer": 900,
    "rebind-timer": 1800,
    "valid-lifetime": 3600,
    // Enable DDNS - Kea will dynamically update the DNS
    "ddns-send-updates" : true,
    "ddns-qualifying-suffix": "your.domain.tld",
    "dhcp-ddns" : {
      "enable-updates": true
    },
    "subnet4": [
      {
        "id": 1001,    // Each subnet requires a unique numeric id
        "subnet": "192.168.56.0/24",
        "pools": [ { "pool": "192.168.56.16 - 192.168.56.254" } ],
        "option-data": [
          {
            "name": "domain-name",
            "data": "your.domain.tld"
          },
          {
            "name": "domain-name-servers",
            "data": "192.168.56.2, 192.168.3.7"
          },
          {
            "name": "domain-search",
            "data": "your.domain.tld"
          },
          {
            "name": "routers",
            "data": "192.168.56.2"
          }
        ]
      }
    ],
    "loggers": [
      {
        "name": "kea-dhcp4",
        "output_options": [
          {
            "output": "/var/log/kea-dhcp4.log",
            "pattern": "%D(%Y-%m-%d %H:%M:%S.%q) %-5p %m\n"
          }
        ],
        "severity": "INFO",
        "debuglevel": 0
      }
    ]
  }
}
```

```
    }
}

// End /etc/kea/kea-dhcp4.conf
EOF
```

IPv6 DHCP Server Configuration

The configuration for IPv6 is similar to the configuration of IPv4. The configuration file is `/etc/kea/kea-dhcp6.conf`.

Dynamic DNS Configuration

If there is a [BIND-9.20.0](#) server running, ISC Kea can update the DNS when it gives an IP address to a client. A sample configuration file is created in `/etc/kea/kea-dhcp-ddns.conf`. Adjust the file to suit your needs or overwrite it by running the following command as the `root` user:

```
cat > /etc/kea/kea-dhcp-ddns.conf << "EOF"
// Begin /etc/kea/kea-dhcp-ddns.conf
{
    "DhcpDdns": {
        "ip-address": "127.0.0.1",
        "port": 53001,
        "control-socket": {
            "socket-type": "unix",
            "socket-name": "/run/kea-ddns-ctrl-socket"
        },
        "tsig-keys": [
            {
                "name": "rndc-key",
                "algorithm": "hmac-sha256",
                "secret": "1FU5hD7faYaaajQCjSdA54JkTPQxbbPrRnzOKqHcd9cM="
            }
        ],
        "forward-ddns" : {
            "ddns-domains" : [
                {
                    "name" : "your.domain.tld.",
                    "key-name": "rndc-key",
                    "dns-servers" : [
                        {
                            "ip-address" : "127.0.0.1",
                            "port" : 53
                        }
                    ]
                }
            ],
            "reverse-ddns" : {
                "ddns-domains" : [
                    {
                        "name" : "56.168.192.in-addr.arpa.",
                        "key-name": "rndc-key",
                        "dns-servers" : [
                            {
                                "ip-address" : "127.0.0.1",
                                "port" : 53
                            }
                        ]
                    }
                ],
                "loggers": [
                    {
                        "name": "kea-dhcp-ddns",
                        "output_options": [
                            {
                                "output": "/var/log/kea-ddns.log",
                                "pattern": "%D(%Y-%m-%d %H:%M:%S.%q) %-5p %m\n"
                            }
                        ],
                        "severity": "INFO",
                    }
                ]
            }
        }
    }
}
```

```

        "debuglevel": 0
    }
}
}
// End /etc/kea/kea-dhcp-ddns.conf
EOF

```

Note

The value of `secret` is just an example. Generate the key for your installation by using the `rndc-confgen -a` command or the `tsig-keygen` command which both are provided by [BIND-9.20.0](#).

In this example configuration, it is assumed that the DNS server runs on the same machine as Kea does (accessible via `127.0.0.1`) and that this machine has the IP `192.168.56.2`.

Contents

Installed Programs: `keactrl`, `kea-admin`, `kea-ctrl-agent`, `kea-dhcp4`, `kea-dhcp6`, `kea-dhcp-ddns`, `kea-lfc`, `kea-shell`

Installed Libraries: `libkea-asiodns.so`, `libkea-asiolink.so`, `libkea-cc.so`, `libkea-cgfclient.so`, `libkea-cryptolink.so`, `libkea-d2srv.so`, `libkea-database.so`, `libkea-dhcp_ddns.so`, `libkea-dhcp++.so`, `libkea-dhcpsrv.so`, `libkea-dns++.so`, `libkea-eval.so`, `libkea-exceptions.so`, `libkea-hooks.so`, `libkea-http.so`, `libkea-log.so`, `libkea-process.so`, `libkea-stats.so`, `libkea-tcp.so`, `libkea-util.so`, and `libkea-util-io.so`

Installed Directories: `/etc/kea`, `/usr/include/kea`, `/usr/lib/kea`, `/usr/lib/python3.12/site-packages/kea`, `/usr/share/kea`, `/usr/share/doc/kea-2.6.1`, and `/var/lib/kea`

Short Descriptions

<code>keactrl</code>	Tool to control (start/stop) the server processes.
<code>kea-admin</code>	<code>kea-admin</code> is a shell script which offers database maintenance.
<code>kea-ctrl-agent</code>	Daemon which exposes a RESTful control interface for managing Kea servers.
<code>kea-dhcp4</code>	The server daemon providing IPv4 addresses.
<code>kea-dhcp6</code>	The server daemon providing IPv6 addresses.
<code>kea-dhcp-ddns</code>	The server daemon performing the dynamic DNS updates.
<code>kea-lfc</code>	The <code>kea-lfc</code> service process removes redundant information from the files used to provide persistent storage for the memfile database backend. It is run by the Kea DHCP server.
<code>keashell</code>	RESTful client to the ISC Kea services.

ProFTPD-1.3.8b

Introduction to ProFTPD

The ProFTPD package contains a secure and highly configurable FTP daemon. This is useful for serving large file archives over a network.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/proftpd/proftpd/archive/v1.3.8b/proftpd-1.3.8b.tar.gz>
- Download MD5 sum: 778cdeeac86e1d26451112bb7d4662af
- Download size: 19 MB
- Estimated disk space required: 66 MB
- Estimated build time: 0.3 SBU

ProFTPD Dependencies

Optional

[libcap-2.70 with PAM](#), [libssh2-1.11.0](#), [Linux-PAM-1.6.1](#), [MariaDB-10.11.8](#) or [MySQL](#), [pcre2-10.44](#), [PostgreSQL-16.4](#), and [Net::SSH2](#)

Installation of ProFTPD

For security reasons, you should install ProFTPD using an unprivileged user and group. As the `root` user:

```
groupadd -g 46 proftpd
useradd -c proftpd -d /srv/ftp -g proftpd \
-s /usr/bin/proftpdshell -u 46 proftpd

install -v -d -m775 -o proftpd -g proftpd /srv/ftp &&
ln -v -s /usr/bin/false /usr/bin/proftpdshell &&
echo /usr/bin/proftpdshell >> /etc/shells
```

Install ProFTPD as an unprivileged user by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc --localstatedir=/run &&
make
```

This packages does not come with a usable test suite.

Now, as the `root` user:

```
make install
install -d -m755 /usr/share/doc/proftpd-1.3.8b &&
cp -Rv doc/* /usr/share/doc/proftpd-1.3.8b
```

Command Explanations

`install -v -d -m775 -o proftpd -g proftpd /srv/ftp`: Create the home directory for ProFTPD.

`ln -v -s /usr/bin/false /usr/bin/proftpdshell`: Set the default shell as a link to an invalid shell.

`echo /usr/bin/proftpdshell >> /etc/shells`: Fake a valid shell for compatibility purposes.

Note

The above two commands can be omitted if the following directive is placed in the configuration file:

```
RequireValidShell off
```

By default, proftpd will require that users logging in have valid shells. The `RequireValidShell` directive turns off this requirement. This is only recommended if you are setting up your FTP server exclusively for anonymous downloads.

Note

Support for most of the dependency packages requires using options passed to the `configure` Script. View the output from `./configure --help` for complete information about enabling dependency packages.

Configuring ProFTPD

Config Files

`/etc/proftpd.conf`

Configuration Information

This is a simple, download-only sample configuration. See the ProFTPD documentation in `/usr/share/doc/proftpd` and consult the website at <http://www.proftpd.org/> for example configurations.

```
cat > /etc/proftpd.conf << "EOF"
# This is a basic ProFTPD configuration file
# It establishes a single server and a single anonymous login.
```

```

ServerName          "ProFTPD Default Installation"
ServerType         standalone
DefaultServer      on

# Port 21 is the standard FTP port.
Port               21
# Umask 022 is a good standard umask to prevent new dirs and files
# from being group and world writable.
Umask              022

# To prevent DoS attacks, set the maximum number of child processes
# to 30. If you need to allow more than 30 concurrent connections
# at once, simply increase this value. Note that this ONLY works
# in standalone mode, in inetd mode you should use an inetd server
# that allows you to limit maximum number of processes per service

MaxInstances       30

# Set the user and group that the server normally runs at.
User               proftpd
Group              proftpd

# To cause every FTP user to be "jailed" (chrooted) into their home
# directory, uncomment this line.
#DefaultRoot ~

# Normally, files should be overwritable.
<Directory /*>
  AllowOverwrite    on
</Directory>

# A basic anonymous configuration, no upload directories.
<Anonymous ~proftpd>
  User             proftpd
  Group            proftpd
  # Clients should be able to login with "anonymous" as well as "proftpd"
  UserAlias        anonymous proftpd

  # Limit the maximum number of anonymous logins
  MaxClients      10

  # 'welcome.msg' should be displayed at login, and '.message' displayed
  # in each newly chdired directory.
  DisplayLogin     welcome.msg
  DisplayChdir     .message

  # Limit WRITE everywhere in the anonymous chroot
  <Limit WRITE>
    DenyAll
  </Limit>
</Anonymous>
EOF

```

Systemd Unit

Install the `proftpd.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-proftpd
```

Contents

Installed Programs: `ftppasswd`, `ftpcount`, `ftpctl`, `ftpmail`, `ftpquota`, `ftpscrub`, `ftpshut`, `ftptop`, `ftpwho`, `in.proftpd` (symlink to `proftpd`), `proftpd`, and `prxs`

Installed Libraries: None

Installed Directory: `/usr/{include,lib}/proftpd`, `/usr/share/doc/proftpd-1.3.8b`, and `/srv/ftp`

Short Descriptions

<code>proftpd</code>	is the FTP daemon
<code>ftpcount</code>	shows the current number of connections

<code>ftpdctl</code>	is used to control the proftpd daemon while it is running
<code>ftpsswd</code>	is a Perl script designed to create and manage AuthUserFiles and AuthGroupFiles of the correct format for proftpd
<code>ftpmail</code>	is a Perl script for sending email based on the proftpd TransferLog
<code>ftpquota</code>	is a Perl script designed to create and manage limits and tally files for the mod_quotatab + mod_quotatab_file module combination for proftpd
<code>ftpscrub</code>	provides a way to scrub the scoreboard file on demand
<code>ftpshut</code>	shuts down all proftpd servers at a given time
<code>ftptop</code>	displays running status on connections
<code>ftpwho</code>	shows current process information for each session
<code>prxs</code>	is a Perl script designed to compile and install third-party modules, from source code, as DSO modules for the installed proftpd

Chapter 21. Mail Server Software

MTAs are the programs which transport mail from one machine to the other. The traditional MTA is Sendmail, however there are several other choices.

In addition to the SMTP servers, there is also a POP/IMAP server (Dovecot).

Dovecot-2.3.21.1

Introduction to Dovecot

Dovecot is an Internet Message Access Protocol (IMAP) and Post Office Protocol (POP) server, written primarily with security in mind. Dovecot aims to be lightweight, fast and easy to set up as well as highly configurable and easily extensible with plugins.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.dovecot.org/releases/2.3/dovecot-2.3.21.1.tar.gz>
- Download MD5 sum: 8c4f360c7f229e4b4371b3d1953d36da
- Download size: 7.5 MB
- Estimated disk space required: 253 MB
- Estimated build time: 4.8 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/dovecot-2.3.21.1-openssl3_fixes-1.patch
- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/dovecot-2.3.21.1-security_fix-1.patch

Dovecot Dependencies

Required

[libtirpc-1.3.5](#)

Optional

[Clucene-2.3.3.4](#), [ICU-75.1](#), [libcap-2.70](#) with PAM, [libunwind-1.6.2](#), [Linux-PAM-1.6.1](#), [Lua-5.4.7](#), [MariaDB-10.11.8](#) or [MySQL](#), [OpenLDAP-2.6.8](#), [PostgreSQL-16.4](#), [SQLite-3.46.1](#), [Valgrind-3.23.0](#), [xfsprogs-6.9.0](#), [Cassandra](#), [stemmer](#) and [libsodium](#)

Installation of Dovecot

There should be dedicated users and groups for unprivileged Dovecot processes and for processing users' logins. Issue the following commands as the `root` user:

```
groupadd -g 42 dovecot &&
useradd -c "Dovecot unprivileged user" -d /dev/null -u 42 \
```

```
-g dovecot -s /bin/false dovecot &&
groupadd -g 43 dovenull &&
useradd -c "Dovecot login user" -d /dev/null -u 43 \
-g dovenull -s /bin/false dovenull
```

First, apply a patch to fix problems with OpenSSL-3:

```
patch -Np1 -i ../dovecot-2.3.21.1-openssl13_fixes-1.patch
```

Next, apply a patch to fix a security vulnerability:

```
patch -Np1 -i ../dovecot-2.3.21.1-security_fix-1.patch
```

Install Dovecot by running the following commands:

```
CPPFLAGS="-I/usr/include/tirpc" \
LDFLAGS+=" -ltirpc" \
./configure --prefix=/usr \
            --sysconfdir=/etc \
            --localstatedir=/var \
            --docdir=/usr/share/doc/dovecot-2.3.21.1 \
            --disable-static \
make
```

To test the results, issue `make -k check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`CPPFLAGS=... LDFLAGS+=...: build with libtirpc instead of the recently removed RPC code provided by GlibC.`

`--disable-static: This switch prevents installation of static versions of the libraries.`

`--with-ldap: This switch enables OpenLDAP authentication support.`

`--with-pgsql: This switch enables PostgreSQL database support.`

`--with-mysql: This switch enables MySQL database support.`

`--with-sqlite: This switch enables SQLite database support.`

`--with-lucene: This switch enables CLucene full text search support.`

`--with-lua: This switch enables Lua plugin support. This includes a mail and push notification plugin.`

Configuring Dovecot

Config Files

`/etc/dovecot/dovecot.conf, /etc/dovecot/conf.d/*, and /etc/dovecot/local.conf`

Configuration Information

Copy an example configuration, which you can use as a starting point:

```
cp -rv /usr/share/doc/dovecot-2.3.21.1/example-config/* /etc/dovecot
```

The following configuration is a simple proof of concept with IMAP service using local users for authentication and mailbox location. Reading files from the `conf.d` directory is commented out since the included example configuration requires OpenSSL and Linux PAM.

```
sed -i '/^#!include / s/^/#/' /etc/dovecot/dovecot.conf &&
chmod -v 1777 /var/mail &&
cat > /etc/dovecot/local.conf << "EOF"
protocols = imap
ssl = no
# The next line is only needed if you have no IPv6 network interfaces
listen = *
```

```

mail_location = mbox:~/Mail:INBOX=/var/mail/%u
userdb {
    driver = passwd
}
passdb {
    driver = shadow
}
EOF

```

You will definitely want to read the official documentation at <https://wiki2.dovecot.org/> if you plan to use Dovecot in a production environment.

Systemd Unit

To start the `dovecot` daemon at boot, enable the previously installed systemd unit with the following command:

```
systemctl enable dovecot
```

Contents

Installed Programs: `doveadm`, `doveconf`, `dovecot`, `dovecot-sysreport`, and `dsync` (symbolic link)

Installed Libraries: various internal plugins in `/usr/lib/dovecot`

Installed Directories: `/etc/dovecot`, `/usr/{include,lib,libexec,share}/dovecot` and `/usr/share/doc/dovecot-2.3.21.1`

Short Descriptions

<code>doveadm</code>	is the Dovecot administration tool
<code>doveconf</code>	is Dovecot's configuration dumping utility
<code>dovecot</code>	is the IMAP and POP server
<code>dovecot-sysreport</code>	prints system information that is useful to the Dovecot developers when submitting bug reports upstream
<code>dsync</code>	is Dovecot's mailbox synchronization utility

Exim-4.98

Introduction to Exim

The Exim package contains a Mail Transport Agent written by the University of Cambridge, released under the GNU Public License.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.exim.org/pub/exim/exim4/exim-4.98.tar.xz>
- Download MD5 sum: f9f4573ba47402364ea71cb3d791b03c
- Download size: 1.9 MB
- Estimated disk space required: 18 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Additional formats of the documentation (text-based docs are shipped with the sources) can be downloaded by following the links shown at <https://exim.org/docs.html>.

Exim Dependencies

Required

[libnl-2.0.1](#), [File-FcntlLock-0.22](#) and [pcre2-10.44](#)

Optional

TDB (alternative to GDBM, built in LFS), [Cyrus SASI-2.1.28](#), [libidn-1.42](#), [Linux-PAM-1.6.1](#), [MariaDB-10.11.8](#) or [MySQL](#), [OpenLDAP-2.6.8](#), [GnuTLS-3.8.7.1](#), [PostgreSQL-16.4](#), [SQLite-3.46.1](#), a graphical environment, [Heimdal GSSAPI](#), [libspf2](#),

and [OpenDMARC](#)

Installation of Exim

Before building Exim, as the `root` user you should create the group and user `exim` which will run the `exim` daemon:

```
groupadd -g 31 exim &&
useradd -d /dev/null -c "Exim Daemon" -g exim -s /bin/false -u 31 exim
```

Configure Exim with the following commands:

```
sed -e 's,^BIN_DIR.*$,BIN_DIRECTORY=/usr/sbin,' \
-e 's,^CONF.*$,CONFIGURE_FILE=/etc/exim.conf,' \
-e 's,^EXIM_USER.*$,EXIM_USER=exim,' \
-e '# USE_OPENSSL/s,#,' src/EDITME > Local/Makefile &&

printf "USE_GDBM = yes\nDBMLIB = -lgdbm\n" >> Local/Makefile
```

If you want to add Linux PAM support, also run the following commands:

```
sed -i '/# SUPPORT_PAM=yes/s,#,' Local/Makefile
echo "EXTRALIBS=-lpam" >> Local/Makefile
```

Build Exim with the following command:

```
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install           &&
install -v -m644 doc/exim.8 /usr/share/man/man8 &&

install -vdm 755      /usr/share/doc/exim-4.98 &&
cp          -Rv doc/*   /usr/share/doc/exim-4.98 &&

ln -sfv exim /usr/sbin/sendmail           &&
install -v -d -m750 -o exim -g exim /var/spool/exim
```

Command Explanations

`sed -e ... > Local/Makefile`: Most of Exim's configuration options are defined in `Local/Makefile`, which is created from the `src/EDITME` file. This command specifies the minimum set of options. Descriptions for the options are listed below.

`printf ... > Local/Makefile`: Setting those variables allows to use GDBM instead of the default Berkeley DB. Remove this command if you have installed [Berkeley DB](#) (deprecated).

`BIN_DIRECTORY=/usr/sbin`: This installs all of Exim's binaries and scripts in `/usr/sbin`.

`CONFIGURE_FILE=/etc/exim.conf`: This installs Exim's main configuration file in `/etc`.

`EXIM_USER=exim`: This tells Exim that after the daemon no longer needs `root` privileges, the process needs to hand off the daemon to the `exim` user.

`USE_OPENSSL`: Uncommenting `USE_OPENSSL=yes` and `USE_OPNSSL_PC=yes` tells the build system to use OpenSSL, and to find the needed libraries with `pkg-config`.

Uncomment `EXIM_MONITOR`: This allows building the Exim monitor program, which requires X Window System support, and is commented out by default.

`ln -sfv exim /usr/sbin/sendmail`: Creates a link to `sendmail` for applications which need it. Exim will accept most Sendmail command-line options.

`install -v -m750 -o exim -g exim /var/spool/exim`: Since `/var/spool` is owned by root and this version of `exim` drops `root` privileges early, to run as user `exim`, it cannot create the `/var/spool/exim` directory. As a work around, it is created manually.

Adding Additional Functionality

To utilize some or all of the dependency packages, you'll need to modify `Local/Makefile` to include the appropriate directives and parameters to link additional libraries before you build Exim. `Local/Makefile` is heavily commented with instructions on how to do this. Listed below is additional information to help you link these dependency packages or add additional functionality.

If you wish to build and install the `.info` documentation, refer to https://exim.org/exim-html-4.98/doc/html/spec_html/ch04.html#SECTinsinfdoc.

If you wish to build in Exim's interfaces for calling virus and spam scanning software directly from access control lists, uncomment the `WITH_CONTENT_SCAN=yes` parameter and review the information found at https://exim.org/exim-html-4.98/doc/html/spec_html/ch45.html.

To use a backend database other than GDBM, see the instructions at https://exim.org/exim-html-4.98/doc/html/spec_html/ch04.html#SECTdb.

For SSL functionality, see the instructions at https://exim.org/exim-html-4.98/doc/html/spec_html/ch04.html#SECTinssl and https://exim.org/exim-html-4.98/doc/html/spec_html/ch42.html.

For tcpwrappers functionality, see the instructions at https://exim.org/exim-html-4.98/doc/html/spec_html/ch04.html#SECID27.

For information about adding authentication mechanisms to the build, see chapters 33–41 of https://exim.org/exim-html-4.98/doc/html/spec_html/index.html.

For information about linking Linux-PAM, refer to the instructions https://exim.org/exim-html-4.98/doc/html/spec_html/ch11.html#SECTexpcond.

For information about linking database engine libraries used for Exim name lookups, see the instructions at https://exim.org/exim-html-4.98/doc/html/spec_html/ch09.html.

If you wish to add Readline support to Exim when invoked in "test expansion" (`-be`) mode, see the information in the `-be` section of https://exim.org/exim-html-4.98/doc/html/spec_html/ch05.html#id2525974.

You may wish to modify the default configuration and send log files to syslog instead of the default `/var/spool/exim/log` directory. See the information at https://exim.org/exim-html-4.98/doc/html/spec_html/ch-log_files.html.

A wealth of information can be also found at the [Exim Wiki](#).

Configuring Exim

Config Files

`/etc/exim.conf` and `/etc/aliases`

Configuration Information

Review the file `/etc/exim.conf`, and modify any settings to suit your needs. Note that the default configuration assumes that the `/var/mail` directory is world writable, but has the sticky bit set. If you want to use the default configuration, issue as the `root` user:

```
chmod -v a+wt /var/mail
```

A default (nothing but comments) `/etc/aliases` file is installed during the package installation if this file did not exist on your system. Create the necessary aliases and start the Exim daemon using the following commands:

```
cat >> /etc/aliases << "EOF"
postmaster: root
MAILER-DAEMON: root
EOF
/usr/sbin/exim -bd -q15m
```

Note

To protect an existing `/etc/aliases` file, the command above appends these aliases to it. This file should be checked and duplicate aliases removed, if present.

The `/usr/sbin/exim -bd -q15m` command starts the Exim daemon with a 15 minute interval in processing the mail queue. Adjust this parameter to suit your desires.

Linux PAM Configuration

If you have built Exim with Linux PAM support, you need to create a PAM configuration file to get it working correctly with BLFS.

Issue the following command as the `root` user to create the configuration file for Linux PAM:

```
cat > /etc/pam.d/exim << "EOF"
# Begin /etc/pam.d/exim

auth    include system-auth
account include system-account
session include system-session

# End /etc/pam.d/exim
EOF
```

Systemd Unit

To automatically start `exim` at boot, install the `exim.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-exim
```

Contents

Installed Programs: `exicyclog`, `exigrep`, `exim`, `exim-4.98-2`, `exim_checkaccess`, `exim_dbmbuild`, `exim_dumpdb`, `exim_fixdb`, `exim_lock`, `exim_tidydb`, `eximstats`, `exinext`, `exipick`, `exiqgrep`, `exiqsumm`, `exiwhat`, and optionally, `eximon`, `eximon.bin`, and `sendmail` (symlink)

Installed Libraries: None

Installed Directories: `/usr/share/doc/exim-4.98` and `/var/spool/exim`

Short Descriptions

<code>exicyclog</code>	cycles Exim log files
<code>exigrep</code>	searches Exim log files
<code>exim</code>	is a symlink to the <code>exim-4.98-2</code> MTA daemon
<code>exim-4.98-2</code>	is the Exim mail transport agent daemon
<code>exim_checkaccess</code>	states whether a given recipient address from a given host is acceptable or not
<code>exim_dbmbuild</code>	creates and rebuilds Exim databases
<code>exim_dumpdb</code>	writes the contents of Exim databases to the standard output
<code>exim_fixdb</code>	modifies data in Exim databases
<code>exim_lock</code>	locks a mailbox file
<code>exim_tidydb</code>	removes old records from Exim databases
<code>eximstats</code>	generates mail statistics from Exim log files
<code>exinext</code>	queries remote host retry times
<code>exipick</code>	selects messages based on various criteria
<code>exiqgrep</code>	is a utility for selective queue listing
<code>exiqsumm</code>	produces a summary of the messages in the mail queue
<code>exiwhat</code>	queries running Exim processes
<code>eximon</code>	is a start-up shell script for <code>eximon.bin</code> used to set the required environment variables before running the program
<code>eximon.bin</code>	is a monitor program which displays current information in an X window, and also contains a menu interface to Exim's command line administration options

Postfix-3.9.0

Introduction to Postfix

The Postfix package contains a Mail Transport Agent (MTA). This is useful for sending email to other users of your host machine. It can also be configured to be a central mail server for your domain, a mail relay agent or simply a mail delivery agent to your local Internet Service Provider.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ghostarchive.org/postfix/postfix-release/official/postfix-3.9.0.tar.gz>
- Download MD5 sum: 3eda9b945ed6cdf11ef58a731c574a04
- Download size: 4.7 MB
- Estimated disk space required: 208 MB
- Estimated build time: 0.1 SBU (Using parallelism=4)

Postfix Dependencies

Recommended

[Cyrus SASL-2.1.28](#), [libnsl-2.0.1](#), and [Imdb-0.9.31](#)

Optional

[ICU-75.1](#) for Email Address Internationalization (SMTPUTF8) support, [MariaDB-10.11.8](#) or [MySQL](#), [OpenLDAP-2.6.8](#), [pcre2-10.44](#), [PostgreSQL-16.4](#), [SQLite-3.46.1](#), [Berkeley DB](#) (deprecated), and [CDB](#) or [TinyCDB](#)

Note that SQLite, MySQL, PostgreSQL and CDB are only useful if there is a known need for them.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/postfix>

Installation of Postfix

Adding Users and Groups

Before you compile the program, you need to create users and groups that will be expected to be in place during the installation. Add the users and groups with the following commands issued by the `root` user:

```
groupadd -g 32 postfix &&
groupadd -g 33 postdrop &&
useradd -c "Postfix Daemon User" -d /var/spool/postfix -g postfix \
-s /bin/false -u 32 postfix &&
chown -v postfix:postfix /var/mail
```

Configuring the Build

The README files are formatted to be read with a pager like less or more. If you want to use a text editor, make them legible with the following sed:

```
sed -i 's/\x08//g' README_FILES/*
```

The Postfix source tree does not contain a `configure` script, rather the makefile in the top-level directory contains a `makefiles` target that regenerates all the other makefiles in the build tree. If you wish to use additional software such as a database back-end for virtual users, or TLS/SSL authentication, you will need to regenerate the makefiles using one or more of the appropriate `CCARGS` and `AUXLIBS` settings listed below.

Note

For all variants of the `CCARGS` you should ensure that '`-DNO_NIS`' is specified so that the build does not attempt to access an `rpcsvc` header which do not exist in BLFS. If [Berkeley DB](#) (deprecated) is not installed, '`-DNO_DB`' needs to be specified as well.

For more details read the readme files.

Initialize the `CCARGS` and `AUXLIBS` according to the note above:

```
CCARGS="-DNO_NIS -DNO_DB"
AUXLIBS=""
```

Next, the `CCARGS` and the `AUXLIBS` variables are filled with settings depending on the actual system configuration. If an optional package is installed but should *not* be included in the Postfix build, simply skip the corresponding `if [...]; then` scriptlets.

Cyrus-SASL

To use Cyrus-SASL with Postfix, use the following arguments:

```

if [ -r /usr/lib/libsasl2.so ]; then
    CCARGS="$CCARGS -DUSE_SASL_AUTH -DUSE_CYRUS_SASL -I/usr/include/sasl"
    AUXLIBS="$AUXLIBS -lsasl2"
fi

```

LMDB

To use LMDB with Postfix, use the following arguments:

```

if [ -r /usr/lib/liblmdb.so ]; then
    CCARGS="$CCARGS -DHAS_LMDB"
    AUXLIBS="$AUXLIBS -llmdb"
fi

```

OpenLDAP

To use OpenLDAP with Postfix, use the following arguments:

```

if [ -r /usr/lib/libldap.so -a -r /usr/lib/liblber.so ]; then
    CCARGS="$CCARGS -DHAS_LDAP"
    AUXLIBS="$AUXLIBS -lldap -llber"
fi

```

Sqlite

To use Sqlite with Postfix, use the following arguments:

```

if [ -r /usr/lib/libsqlite3.so ]; then
    CCARGS="$CCARGS -DHAS_SQLITE"
    AUXLIBS="$AUXLIBS -lsqlite3 -lpthread"
fi

```

MySQL

To use MySQL with Postfix, use the following arguments:

```

if [ -r /usr/lib/libmysqlclient.so ]; then
    CCARGS="$CCARGS -DHAS_MYSQL -I/usr/include/mysql"
    AUXLIBS="$AUXLIBS -lmysqlclient -lz -lm"
fi

```

PostgreSQL

To use PostgreSQL with Postfix, use the following arguments:

```

if [ -r /usr/lib/libpq.so ]; then
    CCARGS="$CCARGS -DHAS_PGSQL -I/usr/include/postgresql"
    AUXLIBS="$AUXLIBS -lpq -lz -lm"
fi

```

CDB/TinyCDB

To use CDB or TinyCDB with Postfix, use the following arguments:

```

if [ -r </path/to/CDB>/libcdb.a ]; then
    CCARGS="$CCARGS -DHAS_CDB"
    AUXLIBS="$AUXLIBS </path/to/CDB>/libcdb.a"
fi

```

StartTLS Authentication

To use OpenSSL with Postfix, use the following arguments:

```

if [ -r /usr/lib/libssl.so -a -r /usr/lib/libcrypto.so ]; then
    CCARGS="$CCARGS -DUSE_TLS -I/usr/include/openssl/"
    AUXLIBS="$AUXLIBS -lssl -lcrypto"
fi

```

OpenSSL is installed by LFS so it should be always there. The `if ... then` is here just for consistency.

Installing Postfix

There might be duplicate settings in `AUXLIBS` like `-lm` when both, MariaDB and PostgreSQL are used. They do not harm. Install Postfix by running the following commands:

```
make CCARGS="$CCARGS" AUXLIBS="$AUXLIBS" makefiles &&
make
```

This package does not come with a useful test suite.

Now, as the `root` user:

```
sh postfix-install -non-interactive \
  daemon_directory=/usr/lib/postfix \
  manpage_directory=/usr/share/man \
  html_directory=/usr/share/doc/postfix-3.9.0/html \
  readme_directory=/usr/share/doc/postfix-3.9.0/readme
```

Command Explanations

`make makefiles`: This command rebuilds the makefiles throughout the source tree to use the options contained in the `CCARGS` and `AUXLIBS` variables.

`sh postfix-install -non-interactive`: This keeps the install script from asking any questions, thereby accepting default destination directories in all but the few cases. If the `html_directory` and `readme_directory` options are not set then the documentation will not be installed.

`CCARGS="-DNO_EAI ..."`: this will turn off SMTPUTF8 support, for example if the rest of your email address infrastructure cannot handle UTF-8 email addresses and message header values.

Configuring Postfix

Config Files

`/etc/aliases`, `/etc/postfix/main.cf`, and `/etc/postfix/master.cf`

Configuration Information

Create (or append to an existing) `/etc/aliases` with the following command. Change `<LOGIN>` to your non-root login identity so mail addressed to `root` can be forwarded to you. As the `root` user:

```
cat >> /etc/aliases << "EOF"
# Begin /etc/aliases

MAILER-DAEMON:      postmaster
postmaster:          root

root:                <LOGIN>
# End /etc/aliases
EOF
```

To protect an existing `/etc/aliases` file, the above command appends these aliases to it if it exists. This file should be checked and duplicate aliases removed, if present.

The BLFS editors recommend to use LMDB instead of Berkeley DB for Postfix tables. Add three lines into `/etc/postfix/main.cf` to make `postmap` encode the lookup tables in the LMDB format by default and to change the default hash setting of the alias tables:

```
echo 'default_database_type = lmdb'        >> /etc/postfix/main.cf &&
echo 'alias_database = lmdb:/etc/aliases' >> /etc/postfix/main.cf &&
echo 'alias_maps = lmdb:/etc/aliases'       >> /etc/postfix/main.cf
```

To protect your server against the recent SMTP smuggling attacks, additional steps are required. Add two lines into `/etc/postfix/main.cf` to disconnect remote SMTP clients that send bare newlines in the DATA section, while also allowing clients on your network with non-standard SMTP implementations to still send mail:

```
echo 'smtpd_forbid_bare_newline = normalize' >> /etc/postfix/main.cf &&
```

```
echo 'smtpd_forbid_bare_newline_exclusions = $mynetworks' >> /etc/postfix/main.cf
```

Note that if you are following an online tutorial to configure Postfix, the tutorial may refer to a lookup table with `hash:/path/to/lookup_table`. You should replace `hash` with `lmdb` in order to use a lookup table encoded in the LMDB format.

Note

The `/etc/postfix/main.cf` and `/etc/postfix/master.cf` files must be personalized for your system. The `main.cf` file needs your fully qualified hostname. You will find that `main.cf` is self documenting, so load it into your editor to make the changes you need for your situation.

Note

Postfix can also be set up to run in a chroot jail. See the file in the source `examples/chroot-setup/LINUX2` for details.

To ensure that all permissions are set properly, postfix provides a tool which is to be run as the `root` user:

```
/usr/sbin/postfix -c /etc/postfix set-permissions
```

If you have an existing configuration, you can run the `postfix` utility to add any necessary definitions to your existing files. As the `root` user:

```
/usr/sbin/postfix upgrade-configuration
```

Before starting Postfix, you should check that your configuration and file permissions will work properly. Run the following commands as the `root` user to check and start your Postfix server:

```
/usr/sbin/postfix check &&  
/usr/sbin/postfix start
```

Systemd Unit

To automate the running of Postfix at startup, install the `postfix.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-postfix
```

Contents

Installed Programs: `mailq` (symlink), `newaliases` (symlink), `postalias`, `postcat`, `postconf`, `postdrop`, `postfix`, `postkick`, `postlock`, `postlog`, `postmap`, `postmulti`, `postqueue`, `postsuper`, and `sendmail`

Installed Libraries: None

Installed Directories: `/{etc,usr/lib}/postfix`, `/usr/share/doc/postfix-3.9.0` and `/var/{lib,spool}/postfix`

Short Descriptions

<code>mailq</code>	A symlink to <code>sendmail</code>
<code>newaliases</code>	A symlink to <code>sendmail</code>
<code>postalias</code>	is a utility for Postfix alias database maintenance
<code>postcat</code>	Prints the contents of files from the Postfix queue in human readable format
<code>postconf</code>	Displays or changes the value of Postfix configuration parameters
<code>postdrop</code>	Creates a file in the maildrop directory and copies its standard input to the file
<code>postfix</code>	is the Postfix control program
<code>postkick</code>	Sends requests to the specified service over a local transport channel
<code>postlock</code>	Locks a mail folder for exclusive use, and executes commands passed to it
<code>postlog</code>	A Postfix -compatible logging interface for use in, for example, shell scripts
<code>postmap</code>	Creates or queries one or more Postfix lookup tables, or updates an existing one
<code>postmulti</code>	is the Postfix multi-instance manager. It allows a system administrator to manage multiple Postfix instances on a single host

<code>postqueue</code>	The Postfix user interface for queue management
<code>postsuper</code>	The Postfix user interface for superuser queue management
<code>sendmail</code>	is the Postfix to Sendmail compatibility interface

sendmail-8.18.1

Introduction to sendmail

The sendmail package contains a Mail Transport Agent (MTA).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.sendmail.org/sendmail.8.18.1.tar.gz>
- Download MD5 sum: b6b332295b5779036d4c9246f96f673c
- Download size: 2.3 MB
- Estimated disk space required: 21 MB
- Estimated build time: 0.1 SBU

sendmail Dependencies

Required

[OpenLDAP-2.6.8](#) (client)

Recommended

[Cyrus SASL-2.1.28](#)

Optional

[ghostscript-10.03.1](#) (for creating PDF documentation), [Procmail-3.24](#) (the configuration proposed below requires that `procmail` be present at run-time), and [nph](#)

Installation of sendmail

Before building sendmail, create the required user, group and directory with the following commands issued as the `root` user:

```
groupadd -g 26 smmsp
useradd -c "Sendmail Daemon" -g smmsp -d /dev/null \
-s /bin/false -u 26 smmsp
chmod -v 1777 /var/mail
install -v -m700 -d /var/spool/mqueue
```

Note

See the source tree `sendmail/README` file for information on linking optional packages into the build. Use the example below, which adds support for SASL, StartTLS (OpenSSL) and OpenLDAP, as a starting point. Of course, modify it to suit your particular needs.

```
cat >> devtools/Site/site.config.m4 << "EOF"
APPENDDEF(`confENVDEF', ` -D STARTTLS -D SASL -D LDAPMAP -D HASFLOCK')
APPENDDEF(`confLIBS', ` -lssl -lcrypto -lsasl2 -lldap -llber')
APPENDDEF(`confINC_DIRS', ` -I/usr/include/sasl')
EOF
```

Install sendmail with the following commands:

```
cat >> devtools/Site/site.config.m4 << "EOF"
define(`confMANGRP', `root')
define(`confMANOWN', `root')
define(`confSBINGRP', `root')
```

```

define(`confUBINGRP', `root')
define(`confUBINOWN', `root')
EOF

sed -i 's|/usr/man/man!|/usr/share/man/man!|' \
    devtools/OS/Linux      &&

cd sendmail          &&
sh Build            &&
cd ../cf/cf          &&
cp generic-linux.mc sendmail.mc &&
sh Build sendmail.cf

```

This package does not come with a test suite.

Now, as the `root` user:

```

install -v -d -m755 /etc/mail &&
sh Build install-cf &&

cd ../../..          &&
sh Build install    &&

install -v -m644 cf/cf/{submit,sendmail}.mc /etc/mail &&
cp -v -R cf/* /etc/mail           &&

install -v -m755 -d /usr/share/doc/sendmail-8.18.1/{cf,sendmail} &&

install -v -m644 CACerts FAQ KNOWNBUGS LICENSE PGPKEYS README RELEASE_NOTES \
    /usr/share/doc/sendmail-8.18.1 &&

install -v -m644 sendmail/{README,SECURITY,TRACEFLAGS,TUNING} \
    /usr/share/doc/sendmail-8.18.1/sendmail &&

install -v -m644 cf/README /usr/share/doc/sendmail-8.18.1/cf &&

for manpage in sendmail editmap mailstats makemap praliases smrsh
do
    install -v -m644 $manpage/$manpage.8 /usr/share/man/man8
done &&

install -v -m644 sendmail/aliases.5   /usr/share/man/man5 &&
install -v -m644 sendmail/mailq.1    /usr/share/man/man1 &&
install -v -m644 sendmail/newaliases.1 /usr/share/man/man1 &&
install -v -m644 vacation/vacation.1 /usr/share/man/man1

```

Install the sendmail Installation and Operations Guide with the following commands:

Note

Remove `op.pdf` from the `make` and `install` commands below if you don't have Ghostscript installed.

```

cd doc/op
sed -i 's/groff/GROFF_NO_SGR=1 groff/' Makefile &&
make op.txt op.pdf

```

Now, as the `root` user:

```

install -v -d -m755 /usr/share/doc/sendmail-8.18.1 &&
install -v -m644 op.ps op.txt op.pdf /usr/share/doc/sendmail-8.18.1 &&
cd ../../..

```

Command Explanations

`cat > devtools/Site/site.config.m4 << "EOF": This creates a configuration file changing some of the default settings.`

`sed ... devtools/OS/Linux: The site.config.m4 does not honor a change to the man directory, so fix it in the OS definitions.`

`sh Build; sh Build sendmail.cf; sh Build install-cf; sh Build install: Sendmail uses an m4 based build script to create the various Makefiles. These commands build and install the package.`

for manpage in...;do...;done; install ...: The man pages are installed already formatted and `man` displays them somewhat garbled. These commands replace the formatted pages with pages `man` can display properly.

Configuring sendmail

Config Files

/etc/mail/*

Configuration Information

Note

Ensure you have a fully qualified domain name defined in `/etc/hosts` for your system before proceeding.

Create the `/etc/mail/local-host-names` and `/etc/mail/aliases` files using the following commands as the `root` user:

```
echo $(hostname) > /etc/mail/local-host-names
cat > /etc/mail/aliases << "EOF"
postmaster: root
MAILER-DAEMON: root

EOF
# Does not work if there is no database backend compiled in:
#newaliases
```

sendmail's primary configuration file, `/etc/mail/sendmail.cf`, is complex and not meant to be directly edited. The recommended method for changing it is to modify `/etc/mail/sendmail.mc` and various m4 files, then run the `m4` macro processor from within `/etc/mail` as follows:

```
cd /etc/mail &&
m4 m4/cf.m4 sendmail.mc > sendmail.cf
```

A full explanation of the files to modify, and the available parameters can be found in `/etc/mail/README`.

Systemd Unit

To automate the running of sendmail at startup, install the `sendmail.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-sendmail
```

Note

The `-qNm` option to `sendmail`, where N is number of minutes, controls how often sendmail will process the mail queue. A default of 5 minutes is used in the `systemd unit`. Individual workstation users may want to set this as low as 1 minute, large installations handling more mail may want to set it higher.

Contents

Installed Programs: `editmap`, `mailstats`, `makemap`, `praliases`, `sendmail`, `smrsh`, and `vacation`; symlinks to `/usr/sbin/sendmail`: `hoststat`, `mailq`, `newaliases`, and `purgestat`

Installed Libraries: None

Installed Directories: `/etc/mail`, `/usr/share/doc/sendmail-8.18.1`, and `/var/spool/clientmqueue`

Short Descriptions

<code>editmap</code>	queries and edits sendmail map files
<code>hoststat</code>	prints sendmail's persistent host status
<code>mailstats</code>	displays sendmail statistics
<code>mailq</code>	prints a summary of outbound mail messages waiting for delivery
<code>makemap</code>	creates sendmail map files

<code>newaliases</code>	rebuilds <code>/etc/mail/aliases.db</code> from the contents of <code>/etc/mail/aliases</code>
<code>praliases</code>	displays current sendmail aliases
<code>purgestat</code>	causes sendmail to clear (purge) all its host-status information
<code>sendmail</code>	is the sendmail mail transport agent
<code>smrsh</code>	is a restricted shell for sendmail
<code>vacation</code>	is an email auto responder

Chapter 22. Databases

This chapter includes databases that range from single-user read/write to industrial database servers with transaction support. Generally, you will be sent here to satisfy dependencies to other applications although building a SQL server on a base LFS system is entirely possible.

Important Notes About Upgrading Database Server Software

Note

This section is about reinstalling database software when an existing database is in use. It is not applicable for initial installations or if there is no existing database for the package being updated, but users should read through it to become aware of issues that can arise in the future.

Let's start this chapter with a dramatic screenshot of an error that really happened. This error will not occur if you are installing database software for the first time:

```
$ sudo systemctl status postgresql
-- postgresql.service - PostgreSQL database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
   Active: failed (Result: exit-code) since Tue 2021-10-26 17:11:53 CDT; 2min 49s ago
     Process: 17336 ExecStart=/usr/bin/pg_ctl -s -D ${PGROOT}/data start -w -t 120 (code=exited, status=1/FAILURE)
       CPU: 7ms

Oct 26 17:11:53 SRVNAME systemd[1]: Starting PostgreSQL database server...
Oct 26 17:11:53 SRVNAME postgres[17338]: 2021-10-26 17:11:53.420 CDT [17338] FATAL:
          database files are incompatible with server
Oct 26 17:11:53 SRVNAME postgres[17338]: 2021-10-26 17:11:53.420 CDT [17338] DETAIL:
          The data directory was initialized by PostgreSQL version 13,
          which is not compatible with this version 14.0.
Oct 26 17:11:53 SRVNAME postgres[17336]: pg_ctl: could not start server
Oct 26 17:11:53 SRVNAME postgres[17336]: Examine the log output.
Oct 26 17:11:53 SRVNAME systemd[1]: postgresql.service: Control process exited, code=exited, status=1/FAILURE
Oct 26 17:11:53 SRVNAME systemd[1]: postgresql.service: Failed with result 'exit-code'.
Oct 26 17:11:53 SRVNAME systemd[1]: Failed to start PostgreSQL database server.
```

To avoid situations like this (i.e., your database server software refuses to start), read the following discussion of the best way to upgrade a DBMS (Database Management System).

The root cause of the error shown above was an upgrade of the server software to a newer major version which left the data files untouched. In this case, the administrator was able to recover the DBMS without any loss of data.

Even if you are doing an initial DBMS install, read through this section. It provides information about implementing backup and restore procedures (or at least a strategy for creating them) which will satisfy your needs and guarantee the safety of your data.

Upgrade Database Server Packages

Database systems work on files which hold the database metadata and the data itself. The internal structure of these files is optimized for use by the server software. When such server software is upgraded, the new software may utilize a different file format than had previously been used. Sometimes the new software can work with the old format as well as the new one—but without the performance improvements the new format provides. Other times, the new server software will reformat the data files automatically after the upgrade.

Unfortunately, the most likely case is that the new server software complains about out of date file formats and exits. When this happens, and you have overwritten the old server software, you may end up with a broken system and lost data.

Changes in data file formats usually happen at major version changes, but they can also occur at other times. Before upgrading any DBMS software, check the documentation to see if this upgrade makes changes which require reformatting the database.

Of course, if you have databases with content that is not easily rebuilt, it is always a good idea to create backups of the database from time to time. Before upgrading the server software, you should run another backup.

Upgrade by Backup and Restore

Note

A backup is useless if there is no verified process to restore the data from this backup. When running a database server, you should not only create backups; you should also verify that the restore process really works. The time to test the restore procedure is *before* you urgently need to recover lost data.

Most database server software provides some basic tools to create backups of your data. Usually the backups created with those tools can be read by newer versions of the software (via a restore tool). Using older restore tools with newer backup data is a bad idea; you should *never* blindly assume that it will work. It might, but usually it doesn't.

The easiest way to upgrade your database files is to

- Create a full database backup using the old tools.

This step creates an offline copy of the database files—for long term archiving, for disaster recovery, or as preparation for an upgrade. This offline backup consists of either (1) a full one-to-one copy of the current database files, or (2) a full backup of the database files from a certain point in time, plus all the journal data (that is Oracle® terminology, it is called "Continuous Archiving" or "write ahead log (WAL)" in Postgresql) describing the changes made after that point in time. This second form takes less time to create (if the DB software provides this type of journaling) because you only have to save the data that have changed since the last full backup was created.

When upgrading database server software, a full backup (which can be used for subsequent incremental backups) should be created; but if there is a lot of data, an incremental backup will suffice. The best strategy for you depends on the amount of data stored in your database (is it a few hundred table rows, or is it hundreds of terabytes?). A full backup in the latter case can't be done quickly. To fully protect your data, create a backup of the old programs (and/or their sources) and save it, along with the data files, to be certain there is a fallback solution if the new software cannot read the old data.

- Upgrade the server software

In this step, instructions to build the database server software are executed just as they are shown in subsequent sections talking about the DBMs like MariaDB or Postgresql. That is, build the software as usual using BLFS instructions.

- Restore the database by using the new tools.

To restore the data, the tools of the newly installed server software should be used. During the restoration process, the new tools will create and/or upgrade the data files in the format the new software requires. It is assumed that newer software is capable of reading old data.

Since you already have a backup procedure in place (and you have tested your restore procedure, right?), this might be the easiest way to upgrade as you can use your well known processes to upgrade just as you always do—at least in terms of the backup and restore.

Upgrade the Database Files by Using System Tools

Some database systems (for instance Postgresql) provide a tool which can reformat (upgrade) the existing database files to the new format. If you need to restore from a backup (for example, running the upgrade tool failed) you will have to reinstall the old software to recover your data.

Even though the reformatting tools might work as advertised, you should create a full backup before running them. A failure could cause serious damage to the database.

Notes for Specific DBMS

PostgreSQL

Upstream documentation for Backup/Restore: <https://www.postgresql.org/docs/current/backup.html>

MariaDB

Upstream documentation for Backup/Restore: <https://mariadb.com/kb/en/backup-and-restore-overview/>

Sqlite

Do not underestimate Sqlite. It is a feature-rich DBMS. The main difference from the two big players above is that Sqlite does not provide access via a network API. Sqlite databases are always stored on the machine running the program which uses the database. The manipulation of data content is done via API calls to library functions directly within the program.

In the upstream documentation you may find the following useful:

Documentation of the sqlite3 command line tool: <https://www.sqlite.org/cli.html>

Documentation of backup API calls: <https://www.sqlite.org/backup.html>

Unfortunately, there is no dedicated chapter in the upstream documentation talking about backup/restore, but there are several articles about it on the Internet. Here is an example.

Documentation for Backup/Restore: <https://database.guide/backup-sqlite-database/>

LMDB

Like Sqlite, this software acts on local database files; there is no network interface.

The relevant resources to back up/restore a LMDB database are the man pages for `mdb_dump` and its counterpart `mdb_load`.

Imdb-0.9.31

Introduction to Imdb

The Imdb package is a fast, compact, key-value embedded data store. It uses memory-mapped files, so it has the read performance of a pure in-memory database while still offering the persistence of standard disk-based databases, and is only limited to the size of the virtual address space

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/LMDB/Imdb/archive/LMDB_0.9.31.tar.gz
- Download MD5 sum: 9d7f059b1624d0a4d4b2f1781d08d600
- Download size: 144 KB
- Estimated disk space required: 4.7 MB
- Estimated build time: less than 0.1 SBU

Installation of Imdb

Note

This package extracts to Imdb-LMDB_0.9.31.

Install Imdb by running the following commands:

```
cd libraries/liblmbd &&
make &&
sed -i 's| liblmbd.a||' Makefile
```

This package does not come with a test suite.

Now, as the `root` user:

```
make prefix=/usr install
```

Command Explanations

`sed ... liblmbd.a ... Makefile`: The package executables use a static library so it must be created. This command suppresses installation of the static library.

Contents

Installed Program: `mdb_copy`, `mdb_dump`, `mdb_load`, and `mdb_stat`

Installed Library: `liblmbd.so`

Installed Directories: None

MariaDB-10.11.8

Introduction to MariaDB

MariaDB is a community-developed fork and a drop-in replacement for the MySQL relational database management system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.mariadb.org/interstitial/mariadb-10.11.8/source/mariadb-10.11.8.tar.gz>
- Download MD5 sum: ba7abfae7947893c5a5343180808b0cb
- Download size: 96 MB
- Estimated disk space required: 1.9 GB (415 MB installed)
- Estimated build time: 4.8 SBU (with parallelism=8, add 0.3 SBU for tests)

Note

The installed size of MariaDB is 725 MB, but this can be reduced by about 265 MB, if desired, by removing the /usr/share/mysql/test directory after installation.

MariaDB Dependencies

Required

[CMake-3.30.2](#)

Recommended

[libevent-2.1.12](#)

Optional

[Boost-1.86.0](#), [libaio-0.3.113](#), [libxml2-2.13.3](#), [Linux-PAM-1.6.1](#), [LZO-2.10](#), [MIT Kerberos V5-1.21.3](#), [pcre2-10.44](#), [Ruby-3.3.4](#), [sphinx-8.0.2](#), [unixODBC-2.3.12](#), [Valgrind-3.23.0](#), [Groonga](#), [KyTea](#), [Judy](#), [MeCab](#), [MessagePack](#), [mruby](#), [MyRocks](#), [Snappy](#), and [ZeroMQ](#)

Installation of MariaDB

Warning

MariaDB and MySQL cannot be installed on the same system without extensive changes to the build configuration of one of the two applications.

For security reasons, running the server as an unprivileged user and group is strongly encouraged. Issue the following (as `root`) to create the user and group:

```
groupadd -g 40 mysql &&
useradd -c "MySQL Server" -d /srv/mysql -g mysql -s /bin/false -u 40 mysql
```

Install MariaDB by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_PREFIX=/usr \
-D GRN_LOG_PATH=/var/log/groonga.log \
-D INSTALL_DOCDIR=share/doc/mariadb-10.11.8 \
-D INSTALL_DOCREADMEDIR=share/doc/mariadb-10.11.8 \
-D INSTALL_MANDIR=share/man \
```

```

-D INSTALL_MYSQLSHAREDIR=share/mysql      \
-D INSTALL_MYSQLTESTDIR=share/mysql/test    \
-D INSTALL_PAMDIR=lib/security             \
-D INSTALL_PAMDATA DIR=/etc/security        \
-D INSTALL_PLUGINDIR=lib/mysql/plugin       \
-D INSTALL_SBINDIR=sbin                     \
-D INSTALL_SCRIPTDIR=bin                   \
-D INSTALL_SQLBENCHDIR=share/mysql/bench   \
-D INSTALL_SUPPORTFILESDIR=share/mysql      \
-D MYSQL_DATADIR=/srv/mysql                \
-D MYSQL_UNIX_ADDR=/run/mysqld/mysqld.sock \
-D WITH_EXTRA_CHARSETS=complex            \
-D WITH_EMBEDDED_SERVER=ON                 \
-D SKIP_TESTS=ON                           \
-D TOKUDB_OK=0                            \
... &&
make

```

To test the results, issue: `make test`. One test, `test-connect`, is known to fail.

Note

A more extensive set of tests can be run with the following:

```

pushd mysql-test
./mtr --parallel <N> --mem --force
popd

```

Where N is the number of tests to run in parallel. Over 5400 tests are run in about 24 SBU with N=4. A few tests may fail, mainly due to character set issues.

Now, as the `root` user:

```
make install
```

If you have [Linux-PAM-1.6.1](#) installed, move the PAM module and configuration file installed by this package as the `root` user:

```

mv -v /usr/share/pam_use_map.so /lib/security &&
mv -v /usr/share/user_map.conf /etc/security

```

Command Explanations

`-D WITH_EMBEDDED_SERVER=ON`: This switch enables compiling the embedded server library needed by certain applications, such as Amarok.

`-D WITH_EXTRA_CHARSETS=complex`: This switch enables support for the complex character sets.

`-D SKIP_TESTS=ON`: This switch disables tests for MariaDB Connector/C which are not supported without additional setup.

`-D WITHOUT_SERVER=ON`: Use this switch if you don't want the server and would like to build the client only.

Note

There are numerous options available to `cmake`. Check the output of the `cmake . -LH` for additional customization options.

Configuring MySQL

Config Files

`/etc/mysql/my.cnf` and `~/.my.cnf`

Configuration Information

Create a basic `/etc/mysql/my.cnf` file using the following command as the `root` user:

```

install -v -dm 755 /etc/mysql &&
cat > /etc/mysql/my.cnf << "EOF"
# Begin /etc/mysql/my.cnf

# The following options will be passed to all MySQL clients
[client]
password      = your_password
port          = 3306
socket        = /run/mysqld/mysqld.sock

# The MySQL server
[mysqld]
port          = 3306
socket        = /run/mysqld/mysqld.sock
datadir       = /srv/mysql
skip-external-locking
key_buffer_size = 16M
max_allowed_packet = 1M
sort_buffer_size = 512K
net_buffer_length = 16K
myisam_sort_buffer_size = 8M

# Don't listen on a TCP/IP port at all.
skip-networking

# required unique id between 1 and 2^32 - 1
server-id     = 1

# Uncomment the following if you are using BDB tables
#bdb_cache_size = 4M
#bdb_max_lock = 10000

# InnoDB tables are now used by default
innodb_data_home_dir = /srv/mysql
innodb_log_group_home_dir = /srv/mysql
# All the innodb_xxx values below are the default ones:
innodb_data_file_path = ibdata1:12M:autoextend
# You can set .._buffer_pool_size up to 50 - 80 %
# of RAM but beware of setting memory usage too high
innodb_buffer_pool_size = 128M
innodb_log_file_size = 48M
innodb_log_buffer_size = 16M
innodb_flush_log_at_trx_commit = 1
innodb_lock_wait_timeout = 50

[mysqldump]
quick
max_allowed_packet = 16M

[mysql]
no-auto-rehash
# Remove the next comment character if you are not familiar with SQL
#safe-updates

[isamchk]
key_buffer = 20M
sort_buffer_size = 20M
read_buffer = 2M
write_buffer = 2M

[myisamchk]
key_buffer_size = 20M
sort_buffer_size = 20M
read_buffer = 2M
write_buffer = 2M

[mysqlhotcopy]
interactive-timeout

# End /etc/mysql/my.cnf
EOF

```

You can now install a database and change the ownership to the unprivileged user and group (perform as the `root` user):

```
mysql_install_db --basedir=/usr --datadir=/srv/mysql --user=mysql &&
chown -R mysql:mysql /srv/mysql
```

Further configuration requires that the MariaDB server is running. Start the server using the following commands as the `root` user:

```
install -v -m755 -o mysql -g mysql -d /run/mysqld &&
mysqld_safe --user=mysql 2>&1 >/dev/null &
```

A default installation does not set up a password for the administrator, so use the following command as the `root` user to set one.

```
mysqladmin -u root password
```

Configuration of the server is now finished. Shut the server down using the following command as the `root` user:

```
mysqladmin -p shutdown
```

Systemd Unit

Install the `mysqld.service` unit included in the [blfs-systemd-units-20240801](#) package as the `root` user to start the MariaDB server during system boot-up.

```
make install-mysqld
```

Note

If you have an existing database already and this installation of binaries was just an upgrade to a newer version, check the upstream documentation for upgrading. It is recommended to run:

```
mariadb-upgrade
```

Contents

Installed Programs: aria_chk, aria_dump_log, aria_ftdump, aria_pack, aria_read_log, aria_s3_copy, galera_new_cluster, galera_recovery, innodb_checksum, mariabackup, mariadb, mariadb-access, mariadb-admin, mariadb-backup, mariadb-binlog, mariadb-check, mariadb-client-test, mariadb-client-test-embedded, mariadb-conv, mariadb-convert-table-format, mariadb-find-rows, mariadb-multi, mariadb-safe, mariadb-safe-helper, mariadb-dump, mariadb-dumpsslow, mariadb-embedded, mariadb-fix-extensions, mariadb-hotcopy, mariadb-import, mariadb-install-db, mariadb-ldb, mariadb-plugin, mariadb-secure-installation, mariadb-setpermission, mariadb-service-convert, mariadb-show, mariadb-slap, mariadb-test, mariadb-test-embedded, mariadb-tzinfo-to-sql, mariadb-upgrade, mariadb-waitpid, mariadb_config, mbstream, msql2mysql, my_print_defaults, myisamchk, myisam_ftdump, myisamlog, myisampack, mysql_config, mytop, perror, replace, resolve_stack_dump, resolveip, sst_dump, wsrep_sst_common, wsrep_sst_mariabackup, wsrep_sst_myisql, and wsrep_sst_rsync

Installed Libraries: libmariadbclient.a, libmariadb.so, libmariadb.so, libmysqlclient.a, libmysqlclient_r.a (symbolic links to libmariadbclient.a), libmysqld.{so,a}, libmysqlservices.a, and several under /usr/lib/mysql/plugin/

Installed Directories: /etc/mysql, /usr/{include,lib,share}/mysql, /usr/share/groonga{-normalizer-mysql}, and /usr/share/doc/mariadb-10.11.8

Short Descriptions

There are several symlinks from mysql* to their mariadb counterparts. These are maintained for compatibility.

The Perl DBI modules must be installed for some of the MariaDB support programs to function properly.

Short Descriptions

<code>aria_chk</code>	is used to check, repair, optimize, sort and get information about Aria tables
<code>aria_dump_log</code>	is a tool to dump the contents of Aria log pages
<code>aria_ftdump</code>	displays full-text index information
<code>aria_pack</code>	is a tool to generate compressed, read-only Aria tables
<code>aria_read_log</code>	displays Aria log file contents
<code>aria_s3_copy</code>	copies an aria table to and from AWS S3
<code>galera_new_cluster</code>	bootstraps a new Galera cluster

<code>galera_recovery</code>	recovers data from a Galera cluster
<code>innchecksum</code>	prints checksums for InnoDB files
<code>mariabackup</code>	is an open source backup tool for InnoDB and XtraDB
<code>mariadb</code>	is a simple SQL shell with input line editing capabilities
<code>mariadb</code>	is the MySQL server daemon
<code>mariadb-access</code>	checks the access privileges for a host name, user name, and database combination
<code>mariadb-admin</code>	is a client for performing administrative operations
<code>mariadb-binlog</code>	reads binary log files
<code>mariadb-check</code>	performs table maintenance: It checks, repairs, optimizes, or analyzes tables
<code>mariadb-client-test</code>	is used for testing aspects of the MySQL client API that cannot be tested using <code>mysqltest</code> and its test language
<code>mariadb-client-test-embedded</code>	is a tool to test the client API for the embedded server
<code>mariadb-conv</code>	converts character sets for use with MariaDB
<code>mariadb-convert-table-format</code>	converts the tables in a database to use a particular storage engine
<code>mariadb-multi</code>	is designed to manage several mysqld processes that listen for connections on different Unix socket files and TCP/IP ports
<code>mariadb-safe</code>	is the recommended way to start a mysqld server on Unix and NetWare
<code>mariadb-dump</code>	is a backup program
<code>mariadb-dumpsslow</code>	parses MySQL slow query log files and prints a summary of their contents
<code>mariadb-embedded</code>	is a MySQL client statically linked to libmariadb
<code>mariadb-find-rows</code>	reads files containing SQL statements and extracts statements that match a given regular expression or that contain USE db_name or SET statements
<code>mariadb-fix-extensions</code>	converts the extensions for MyISAM (or ISAM) table files to their canonical forms
<code>mariadb-hotcopy</code>	locks the table, flushes the table and then performs a copy of the database
<code>mariadb-import</code>	reads a range of data formats, and inserts the data into a database
<code>mariadb-install-db</code>	initializes the MySQL data directory and creates the system tables that it contains, if they do not exist
<code>mariadb-ldb</code>	is the RocksDB tool
<code>mariadb-plugin</code>	is a utility that enables MySQL administrators to manage which plugins a MySQL server loads
<code>mariadb-secure-installation</code>	is a tool to improve MySQL installation security
<code>mariadb-service-convert</code>	generates a systemd unit based on the current mariadb settings
<code>mariadb-setpermission</code>	sets permissions in the MySQL grant tables
<code>mariadb-show</code>	shows the structure of a MariaDB database
<code>mariadb-slap</code>	is a diagnostic program designed to emulate client load for a MySQL server and to report the timing of each stage
<code>mariadb-test</code>	runs a test case against a MySQL server and optionally compares the output with a result file
<code>mariadb-test-embedded</code>	is similar to the <code>mysqltest</code> command but is built with support for the libmysqld embedded server
<code>mariadb-tzinfo-to-sql</code>	loads the time zone tables in the mysql database
<code>mariadb-upgrade</code>	examines all tables in all databases for incompatibilities with the current version of MySQL Server
<code>mariadb-waitpid</code>	signals a process to terminate and waits for the process to exit
<code>mariadb_config</code>	gets compiler flags for using the MariaDB Connector/C
<code>mbstream</code>	is an utility for sending InnoDB and XTraDB backups over a stream
<code>mysql2mysql</code>	is a tool to convert mSQL programs for use with MySQL
<code>my_print_defaults</code>	displays the options from option groups of option files
<code>myisam_ftdump</code>	displays information about FULLTEXT indexes in MyISAM tables
<code>myisamchk</code>	gets information about your database tables or checks, repairs, or optimizes them
<code>myisamlog</code>	displays MyISAM log file contents
<code>myisampack</code>	is a tool for compressing MyISAM tables
<code>mysql_config</code>	provides you with useful information for compiling your MySQL client and connecting it to MySQL

<code>mytop</code>	is a console-based tool for monitoring the threads and overall performance of a MySQL server
<code>perror</code>	is a utility that displays descriptions for system or storage engine error codes
<code>replace</code>	is a MariaDB/MySQL extension to the SQL standard
<code>resolve_stack_dump</code>	resolves a numeric stack dump to symbols
<code>resolveip</code>	is a utility for resolving IP addresses to host names and vice versa
<code>sst_sump</code>	dumps the content of sst files (the format used by RocksDB)

PostgreSQL-16.4

Introduction to PostgreSQL

PostgreSQL is an advanced object-relational database management system (ORDBMS), derived from the Berkeley Postgres database management system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.postgresql.org/pub/source/v16.4/postgresql-16.4.tar.bz2>
- Download MD5 sum: bdcc1e350b473c13d132d190c0c18499
- Download size: 24 MB
- Estimated disk space required: 225 MB (add 43 MB for tests)
- Estimated build time: 0.8 SBU (with parallelism=4, add 0.2 SBU for tests)

PostgreSQL Dependencies

Optional

[ICU-75.1](#), [libxml2-2.13.3](#), [libxslt-1.1.42](#), [OpenLDAP-2.6.8](#), [Linux-PAM-1.6.1](#), [MIT Kerberos V5-1.21.3](#) and [Bonjour](#)

Optional (To Regenerate Documentation)

[fop-2.9](#), [docbook-4.5-dtd](#), [docbook-dsssl-1.79](#), [DocBook-utils-0.6.14](#), [OpenJade-1.3.2](#), and [SGMLSp-1.1](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/postgresql>

Installation of PostgreSQL

For enhanced security, it is better to have a dedicated group and user for running the PostgreSQL server. First, issue as the `root` user:

```
groupadd -g 41 postgres &&
useradd -c "PostgreSQL Server" -g postgres -d /srv/pgsql/data \
-u 41 postgres
```

Note

There are several configuration items that add additional functionality with optional packages to PostgreSQL. Use `./configure --help` to see a list.

Install PostgreSQL with the following commands:

```
sed -i '/DEFAULT_PGSOCKET_DIR/s@/tmp@/run/postgresql@' src/include/pg_config_manual.h &&
./configure --prefix=/usr           \
            --enable-thread-safety \
            --docdir=/usr/share/doc/postgresql-16.4 &&
make
```

There are a number of programs in the `contrib/` directory. If you are going to run this installation as a server and wish to build some of them, enter `make -C contrib` OR `make -C contrib/<SUBDIR-NAME>` for each subdirectory.

Tests must be run as an unprivileged user because they need to start a temporary server and this is prevented as the root user. For the same reason, you need to stop all PostgreSQL servers if any are running. If a previous version of PostgreSQL is installed, it may be necessary to use `--disable-rpath` with `configure` to avoid failures, but *installing the binaries created using this switch is not recommended*. To test the results, issue: `make check`.

Note

If you are installing PostgreSQL to upgrade an existing installation, there are important steps that you need to follow. If the major version of the new build is greater than the previous version, there is a chance that the data file format has changed. New software cannot use the existing data files. In this case, the server will not start because the old programs have been overwritten, so the data is unavailable until its file format has been converted.

Before upgrading an existing installation of PostgreSQL, check the documentation for any considerations that you must keep in mind during the upgrade. Note that new major versions might use a different binary format in the data objects, causing potential incompatibilities. For more information, please review upstream's documentation about upgrading PostgreSQL here:

<https://www.postgresql.org/docs/current/upgrading.html>.

At this point, you may have both the old and the new binaries installed on your filesystem. These binaries can be used to perform an upgrade of your existing database files. For the following instructions it is assumed that

- The actual data files are stored in `/srv/pgsql/data`
- The upgraded data files will be stored in `/srv/pgsql/newdata`
- There is enough disk space to hold the actual data files twice. The upgrade is not an inline upgrade but it will copy the data to new database files.

First, do a temporary install which makes access to the new binaries much easier:

```
make DESTDIR=$(pwd)/DESTDIR install
```

Next, create a directory which is writable by the `postgres` user, as the `root` user:

```
install -d -o postgres $(pwd)/DESTDIR/tmp
```

Now, stop the existing instance of PostgreSQL and start the upgrade process as the `root` user:

```
pushd $(pwd)/DESTDIR/tmp  
systemctl stop postgresql  
su postgres -c ".../usr/bin/initdb -D /srv/pgsql/data"  
su postgres -c ".../usr/bin/pg_upgrade \  
          -d /srv/pgsql/data      -b /usr/bin \  
          -D /srv/pgsql/newdata -B .../usr/bin"  
popd
```

At this point, your database files are available in two locations on disk. The old data is located in `/srv/pgsql/data`, and the new data is in `/srv/pgsql/newdata`. Backing up the old database files is recommended before continuing.

Next, remove the old database files, and rename the new data directory as the `root` user:

```
rm -rf /srv/pgsql/data  
mv /srv/pgsql/newdata /srv/pgsql/data
```

Now, as the `root` user:

```
make install      &&  
make install-docs
```

If you made any of the `contrib/` programs, as the `root` user:

```
make -C contrib/<SUBDIR-NAME> install
```

Tip

If you only intend to use PostgreSQL as a client to connect to a server on another machine, your installation is complete and you should not run the remaining commands.

If you have upgraded an existing database, skip the rest of the commands because your database is ready to use. If this is the first time you install PostgreSQL, continue with the initialization.

Initialize a database cluster with the following commands issued by the `root` user:

```
install -v -dm700 /srv/pgsql/data &&
install -v -dm755 /run/postgresql &&
chown -Rv postgres:postgres /srv/pgsql /run/postgresql
```

Now, initialize the database as the `root` user:

```
su - postgres -c '/usr/bin/initdb -D /srv/pgsql/data'
```

Command Explanations

`sed -i ...`: This sed changes the server socket location from `/tmp` to `/run/postgresql`.

`--enable-thread-safety`: This switch makes the client libraries thread-safe by allowing concurrent threads in `libpq` and ECPG programs to safely control their private connection handles.

`--with-openssl`: builds the package with support for OpenSSL encrypted connections.

`--with-perl`: builds the PL/Perl server-side language.

`--with-python`: builds the PL/Python server-side language.

`--with-tcl`: builds the PL/Tcl server-side language.

Configuring PostgreSQL

Config Files

`$PGDATA/pg_ident.conf`, `$PGDATA/pg_hba.conf`, and `$PGDATA/postgresql.conf`

The `PGDATA` environment variable is used to distinguish database clusters from one another by setting it to the value of the directory which contains the cluster desired. The three configuration files exist in every `PGDATA/` directory. Details on the format of the files and the options that can be set in each can be found in [/usr/share/doc/postgresql-16.4/html/index.html](#).

Systemd Unit

Install the `postgresql.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-postgresql
```

Starting the PostgreSQL Server and Creating a Sample Database

The database server can be manually started with the following command (as the `root` user):

```
su - postgres -c '/usr/bin/postgres -D /srv/pgsql/data > \
/srv/pgsql/data logfile 2>&1 &'
```

Note

If you are scripting this part, you should wait for the server to start before going on, by adding for example `sleep 2` after the above command.

The instructions below show how to create a database, add a table to it, insert some rows into the table and select them, to verify that the installation is working properly. Still as user `root`, issue:

```
su - postgres -c '/usr/bin/createdb test' &&
echo "create table t1 ( name varchar(20), state_province varchar(20) );" \
| (su - postgres -c '/usr/bin/psql test') &&
echo "insert into t1 values ('Billy', 'NewYork');" \
| (su - postgres -c '/usr/bin/psql test') &&
```

```

echo "insert into t1 values ('Evanidus', 'Quebec');" \
| (su - postgres -c '/usr/bin/psql test') &&
echo "insert into t1 values ('Jesse', 'Ontario');" \
| (su - postgres -c '/usr/bin/psql test') &&
echo "select * from t1;" | (su - postgres -c '/usr/bin/psql test')

```

When you are done with testing, you can shut down the server, by issuing as `root`:

```
su - postgres -c "/usr/bin/pg_ctl stop -D /srv/pgsql/data"
```

Contents

- Installed Programs:** `clusterdb`, `createdb`, `createuser`, `dropdb`, `dropuser`, `ecpg`, `initdb`, `pg_amcheck`, `pg_archivecleanup`, `pg_basebackup`, `pg_checksums`, `pg_config`, `pg_controldata`, `pg_ctl`, `pg_dump`, `pg_dumpall`, `pg_isready`, `pg_recvwal`, `pg_recvlogical`, `pg_resetwal`, `pg_restore`, `pg_rewind`, `pg_test_fsync`, `pg_test_timing`, `pg_upgrade`, `pg_verifybackup`, `pg_waldump`, `pgbench`, `postgres`, `psql`, `reindexdb`, `vacuumdb`, optionally, if Tcl support has been built, `pltcl_delmod`, `pltcl_listmod`, `pltcl_loadmod`, and optionally (in contrib/) `oid2name`, `pg_standby`, `vacuumlo`, and many others
- Installed Libraries:** `libecpg.{so,a}`, `libecpg_compat.{so,a}`, `libpgcommon.a`, `libpgcommon_shlib.a`, `libpgfeutils.a`, `libpgport.a`, `libpgport_shlib.a`, `libpgtypes.{so,a}`, `libpq.{so,a}`, various charset modules and optionally programming language modules under `/usr/lib/postgresql`
- Installed Directories:** `/usr/include/{libpq,postgresql}`, `/usr/lib/postgresql`, `/usr/share/{doc/postgresql-16.4,postgresql}`, and `/srv/pgsql`

Short Descriptions

<code>clusterdb</code>	is a utility for reclustering tables in a PostgreSQL database
<code>createdb</code>	creates a new PostgreSQL database
<code>createuser</code>	defines a new PostgreSQL user account
<code>dropdb</code>	removes a PostgreSQL database
<code>dropuser</code>	removes a PostgreSQL user account
<code>ecpg</code>	is the embedded SQL preprocessor
<code>initdb</code>	creates a new database cluster
<code>oid2name</code>	resolves OIDs (Object IDs) and file nodes in a PostgreSQL data directory
<code>pg_amcheck</code>	checks for corruption in one or more PostgreSQL databases
<code>pg_archivecleanup</code>	cleans up PostgreSQL WAL (write-ahead log) archive files
<code>pg_basebackup</code>	takes base backups of a running PostgreSQL cluster
<code>pg_checksums</code>	enables, disables, or checks data checksums in a PostgreSQL database cluster
<code>pg_config</code>	retrieves PostgreSQL version information
<code>pg_controldata</code>	returns information initialized during <code>initdb</code> , such as the catalog version and server locale
<code>pg_ctl</code>	controls stopping and starting the database server
<code>pg_dump</code>	dumps database data and metadata into scripts which are used to recreate the database
<code>pg_dumpall</code>	recursively calls <code>pg_dump</code> for each database in a cluster
<code>pg_isready</code>	checks the connection status of a PostgreSQL server
<code>pg_recvwal</code>	is used to stream write-ahead logs from a PostgreSQL server
<code>pg_recvlogical</code>	controls PostgreSQL logical decoding streams
<code>pg_resetwal</code>	resets the write-ahead log and other control information of a PostgreSQL database cluster
<code>pg_restore</code>	creates databases from dump files created by <code>pg_dump</code>
<code>pg_rewind</code>	synchronizes a PostgreSQL data directory with another data directory that was forked from the first one
<code>pg_standby</code>	supports the creation of a PostgreSQL warm standby server
<code>pg_test_fsync</code>	determines the fastest wal_sync method for PostgreSQL
<code>pg_test_timing</code>	measures timing overhead
<code>pg_upgrade</code>	upgrades a PostgreSQL server instance
<code>pg_verifybackup</code>	verifies the integrity of a base backup of a PostgreSQL cluster
<code>pg_waldump</code>	displays a human-readable rendering of the write-ahead log of a PostgreSQL database cluster
<code>pgbench</code>	runs a benchmark test on PostgreSQL
<code>pltcl_delmod</code>	is a support script used to delete a module from a PL/Tcl table. The command requires the PgTcl package to be installed

<code>pltcl_listmod</code>	is a support script used to list the modules in a PL/Tcl table. The command requires the Pgtcl package to be installed
<code>pltcl_loadmod</code>	is a support script used to load a module into a PL/Tcl table. The command requires the Pgtcl package to be installed
<code>postgres</code>	is the PostgreSQL database server
<code>psql</code>	is a console based database shell
<code>reindexdb</code>	is a utility for rebuilding indexes in a database
<code>vacuumdb</code>	compacts databases and generates statistics for the query analyzer
<code>vacuumlo</code>	removes orphaned large objects from a PostgreSQL database
<code>libecpg.{so,a}</code>	contains functions to support embedded SQL in C programs
<code>libecpg_compat.{so,a}</code>	is the ecpg compatibility library
<code>libgport.a</code>	is the port-specific subsystem of the Postgres backend
<code>libpgtypes.{so,a}</code>	contains functions for dealing with Postgres data types
<code>libpq.{so,a}</code>	is the C programmer's API to Postgres

SQLite-3.46.1

Introduction to SQLite

The SQLite package is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sqlite.org/2024/sqlite-autoconf-3460100.tar.gz>
- Download MD5 sum: bde7df8c2a4bce82dbf42f7e4dda0e21
- Download size: 3.1 MB
- Estimated disk space required: 80 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

Additional Downloads

Optional Documentation

- Download (HTTP): <https://sqlite.org/2024/sqlite-doc-3460100.zip>
- Download MD5 sum: 3f1fc03c5aad4839825d576eea5d93ff
- Download size: 11 MB

SQLite Dependencies

Optional

[libedit](#) and [UnZip-6.0](#) (required to unzip the documentation)

Installation of SQLite

If you downloaded the optional documentation, issue the following command to install the documentation into the source tree:

```
unzip -q ../sqlite-doc-3460100.zip
```

Install SQLite by running the following commands:

```
./configure --prefix=/usr \
--disable-static \
--enable-fts{4,5} \
CPPFLAGS="-D SQLITE_ENABLE_COLUMN_METADATA=1 \
-D SQLITE_ENABLE_UNLOCK_NOTIFY=1 \
-D SQLITE_ENABLE_DBSTAT_VTAB=1 \

```

```
-D SQLITE_SECURE_DELETE=1" &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you downloaded the optional documentation, issue the following commands as the `root` user to install it:

```
install -v -m755 -d /usr/share/doc/sqlite-3.46.1 &&  
cp -v -R sqlite-doc-3460100/* /usr/share/doc/sqlite-3.46.1
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-fs{4,5}`: These switches enable support for version 3, 4 and 5 of the full text search (FTS) extension. Note that the `--enable-fs4` switch enables support for both FTS version 4 and FTS version 3.

`CPPFLAGS="-D SQLITE_ENABLE_COLUMN_METADATA=1 ..."`: Some applications require these options to be turned on. The only way to do this is to include them in the `CFLAGS` or `CPPFLAGS`. We use the latter so the default value (or any value set by the user) of `CFLAGS` won't be affected. For further information on what can be specified see <https://www.sqlite.org/compile.html>.

Contents

Installed Program: `sqlite3`

Installed Library: `libsqllite3.so`

Installed Directory: `/usr/share/doc/sqlite-3.46.1`

Short Descriptions

<code>sqlite3</code>	is a terminal-based front-end to the SQLite library that can evaluate queries interactively and display the results
<code>libsqllite3.so</code>	contains the SQLite API functions

Chapter 23. Other Server Software

Here you will find many ways to share your machine with the rest of the world or your local network. Before installing any packages in this chapter, you need to be sure you understand what the package does and how to set it up correctly. It might also be helpful to learn about the consequences of an improper setup so that you can analyze the risks.

OpenLDAP-2.6.8

Introduction to OpenLDAP

The OpenLDAP package provides an open source implementation of the Lightweight Directory Access Protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.openldap.org/software/download/OpenLDAP/openldap-release/openldap-2.6.8.tgz>
- Download MD5 sum: a7ca5f245340e478ea18b8f972c89bb1
- Download size: 6.3 MB
- Estimated disk space required: 61 MB (client and server)
- Estimated build time: 0.4 SBU (client), 1.1 SBU (server)

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/openldap-2.6.8-consolidated-1.patch>

OpenLDAP Dependencies

Recommended

[Cyrus SASL-2.1.28](#)

Optional

[GnuTLS-3.8.7.1](#), [unixODBC-2.3.12](#), [MariaDB-10.11.8](#) or [PostgreSQL-16.4](#) or [MySQL](#), [OpenSLP](#), [WiredTiger](#), and [Berkeley DB](#) (deprecated) (for slapd, also deprecated)

Installation of OpenLDAP

Note

If you only need to install the client side `ldap*` binaries, corresponding man pages, libraries and header files (referred to as a “client-only” install), issue these commands instead of the following ones (no test suite available):

```
patch -Np1 -i ../openldap-2.6.8-consolidated-1.patch &&
autoconf &&

./configure --prefix=/usr      \
--sysconfdir=/etc      \
--disable-static      \
--enable-dynamic      \
--disable-debug      \
--disable-slapd &&

make depend &&
make
```

Then, as the `root` user:

```
make install
```

There should be a dedicated user and group to take control of the `slapd` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 83 ldap &&
useradd -c "OpenLDAP Daemon Owner" \
-d /var/lib/openldap -u 83 \
-g ldap -s /bin/false ldap
```

Install OpenLDAP by running the following commands:

```
patch -Np1 -i ../openldap-2.6.8-consolidated-1.patch &&
autoconf &&

./configure --prefix=/usr      \
--sysconfdir=/etc      \
--localstatedir=/var      \
--libexecdir=/usr/lib      \
--disable-static      \
--disable-debug      \
--with-tls=openssl      \
--with-cyrus-sasl      \
--without-systemd      \
--enable-dynamic      \
--enable-crypt      \
--enable-spasswd      \
--enable-slapd      \
--enable-modules      \
--enable-rlookups      \
--enable-backends=mod \
--disable-sql      \
--disable-wt      \
--enable-overlays=mod &&
```

```
make depend &&
make
```

The tests are fragile, and errors may cause the tests to abort prior to finishing. Some errors may happen due to timing problems. The tests take around an hour, and the time is CPU independent due to delays in the tests. On most systems, the tests will run up to the `test065-proxyauth` for `ldb` test. To test the results, issue: `make test`.

Now, as the `root` user:

```
make install &&

sed -e "s/\.la/.so/" -i /etc/openldap/slapd.{conf,ldif}{,.default} &&

install -v -dm700 -o ldap -g ldap /var/lib/openldap      &&

install -v -dm700 -o ldap -g ldap /etc/openldap/slapd.d &&
chmod   -v    640    /etc/openldap/slapd.{conf,ldif}    &&
chown   -v    root:ldap /etc/openldap/slapd.{conf,ldif} &&

install -v -dm755 /usr/share/doc/openldap-2.6.8 &&
cp      -vfr      doc/{drafts/rfc,guide} \
          /usr/share/doc/openldap-2.6.8
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-debug`: This switch disables the debugging code in OpenLDAP.

`--enable-dynamic`: This switch forces the OpenLDAP libraries to be dynamically linked to the executable programs.

`--enable-crypt`: This switch enables using [crypt\(3\)](#) passwords.

`--enable-spasswd`: This switch enables SASL password verification.

`--enable-modules`: This switch enables dynamic module support.

`--enable-rlookups`: This switch enables reverse lookups of client hostnames.

`--enable-backends`: This switch enables all available backends.

`--enable-overlays`: This switch enables all available overlays.

`--disable-sql`: This switch explicitly disables the SQL backend. Omit this switch if a SQL server is installed and you are going to use a SQL backend.

`--disable-wt`: This switch explicitly disables the WiredTiger backend. Omit this switch if WiredTiger is installed and you are going to use a WiredTiger backend.

`--libexecdir=/usr/lib`: This switch controls where the `/usr/lib/openldap` directory is installed. Everything in that directory is a library, so it belongs under `/usr/lib` instead of `/usr/libexec`.

`--enable-slp`: This switch enables SLPv2 support. Use it if you have installed [OpenSLP](#).

`--disable-versioning`: This switch disables symbol versioning in the OpenLDAP libraries. The default is to have symbol versioning. Note that if you have built applications using this package with symbol versioning, and remove the symbols, the applications may fail to run.

Note

You can run `./configure --help` to see if there are other switches you can pass to the `configure` command to enable other options or dependency packages.

`install ...`, `chown ...`, and `chmod ...`: Having `slapd` configuration files and `ldap` databases in `/var/lib/openldap` readable by anyone is a SECURITY ISSUE, especially since a file stores the admin password in PLAIN TEXT. That's why mode 640 and `root:ldap` ownership were used. The owner is `root`, so only `root` can modify the file, and group is `ldap`, so that the group which owns `slapd` daemon could read but not modify the file in case of a security breach.

Configuring OpenLDAP

Config Files

- For LDAP client: `/etc/openldap/ldap.conf` and `~/.ldaprc`
- For LDAP server, two configuration mechanisms are used: a legacy `/etc/openldap/slapd.conf` configuration file and the recommended `slapd-config` system, using an LDIF database stored in `/etc/openldap/slapd.d`.

Configuration Information

Configuring the `slapd` servers can be complex. Securing the LDAP directory, especially if you are storing non-public data such as password databases, can also be a challenging task. In order to set up OpenLDAP, you'll need to modify either the `/etc/openldap/slapd.conf` file (old method), or the `/etc/openldap/slapd.ldif` file and then use `ldapadd` to create the LDAP configuration database in `/etc/openldap/slapd.d` (recommended by the OpenLDAP documentation).

Warning

The instructions above install an empty LDAP structure and a default `/etc/openldap/slapd.conf` file, which are suitable for testing the build and other packages using LDAP. Do not use them on a production server.

Resources to assist you with topics such as choosing a directory configuration, backend and database definitions, access control settings, running as a user other than `root` and setting a `chroot` environment include:

- The [`slapd\(8\)`](#) man page.
- The [`slapd.conf\(5\)`](#) and [`slapd-config\(5\)`](#) man pages.
- The [`OpenLDAP 2.6 Administrator's Guide`](#) (also installed locally in `/usr/share/doc/openldap-2.6.8/guide/admin`).
- Documents located at <https://www.openldap.org/pub/>.

Systemd Unit

To automate the startup of the LDAP server at system bootup, install the `slapd.service` unit included in the [`blfs-systemd-units-20240801`](#) package using the following command:

```
make install-slapd
```

Note

You'll need to modify `/etc/default/slapd` to include the parameters needed for your specific configuration. See the `slapd` man page for parameter information.

Testing the Configuration

Start the LDAP server using `systemctl`:

```
systemctl start slapd
```

Verify access to the LDAP server with the following command:

```
ldapsearch -x -b '' -s base '(objectclass=*)' namingContexts
```

The expected result is:

```
# extended LDIF
#
# LDAPv3
# base <> with scope baseObject
# filter: (objectclass=*)
# requesting: namingContexts
#
#
# dn:
namingContexts: dc=my-domain,dc=com
#
# search result
```

```
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Contents

- Installed Programs:** ldapadd, ldapcompare, ldapdelete, ldapexop, ldapmodify, ldapmodrdn, ldappasswd, ldapsearch, ldapurl, ldapvc, ldapwhoami, slapacl, slapadd, slapauth, slapcat, slapd, slapdn, slapindex, slapmodify, slappasswd, slapschema, and slapttest
- Installed Libraries:** liblber.so, libldap.so, and several under /usr/lib/openldap
- Installed Directories:** /etc/openldap, /{usr,var}/lib/openldap, and /usr/share/doc/openldap-2.6.8

Short Descriptions

ldapadd	opens a connection to an LDAP server, binds and adds entries
ldapcompare	opens a connection to an LDAP server, binds and performs a compare using specified parameters
ldapdelete	opens a connection to an LDAP server, binds and deletes one or more entries
ldapexop	issues the LDAP extended operation specified by oid or one of the special keywords whoami, cancel, or refresh
ldapmodify	opens a connection to an LDAP server, binds and modifies entries
ldapmodrdn	opens a connection to an LDAP server, binds and modifies the RDN of entries
ldappasswd	is a tool used to set the password of an LDAP user
ldapsearch	opens a connection to an LDAP server, binds and performs a search using specified parameters
ldapurl	is a command that allows to either compose or decompose LDAP URIs
ldapvc	verifies LDAP credentials
ldapwhoami	opens a connection to an LDAP server, binds and displays whoami information
slapacl	is used to check the behavior of slapd by verifying access to directory data according to the access control list directives defined in its configuration
slapadd	is used to add entries specified in LDAP Directory Interchange Format (LDIF) to an LDAP database
slapauth	is used to check the behavior of the slapd in mapping identities for authentication and authorization purposes, as specified in slapd.conf
slapcat	is used to generate an LDAP LDIF output based upon the contents of a slapd database
slapd	is the standalone LDAP server
slapdn	checks a list of string-represented DNs based on schema syntax
slapindex	is used to regenerate slapd indexes based upon the current contents of a database
slapmodify	modifies entries in a slapd database
slappasswd	is an OpenLDAP password utility
slapschema	is used to check schema compliance of the contents of a slapd database
slapttest	checks the sanity of the <code>slapd.conf</code> file
liblber.so	is a set of Lightweight Basic Encoding Rules routines. These routines are used by the LDAP library routines to encode and decode LDAP protocol elements using the (slightly simplified) Basic Encoding Rules defined by LDAP. They are not normally used directly by an LDAP application program except in the handling of controls and extended operations
libldap.so	supports the LDAP programs and provide functionality for other programs interacting with LDAP

Unbound-1.21.0

Introduction to Unbound

Unbound is a validating, recursive, and caching DNS resolver. It is designed as a set of modular components that incorporate modern features, such as enhanced security (DNSSEC) validation, Internet Protocol Version 6 (IPv6), and a client resolver library API as an integral part of the architecture.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://nlnetlabs.nl/downloads/unbound/unbound-1.21.0.tar.gz>
- Download MD5 sum: fb7cc7922064bf122941b9f135052d00

- Download size: 6.3 MB
- Estimated disk space required: 148 MB (with docs; add 11 MB for tests)
- Estimated build time: 0.3 SBU (Using parallelism=4; with docs; add 0.3 SBU for tests)

Unbound Dependencies

Optional

[libevent-2.1.12](#), [Nettle-3.10](#), [Protobuf-c-1.5.0](#) (for dnstap), [Python2](#), [sphinx-8.0.2](#) (for Python bindings documentation), [SWIG-4.2.1](#) (for Python bindings), [Doxygen-1.12.0](#) (for html documentation), and [dnstap](#)

Installation of Unbound

There should be a dedicated user and group to take control of the `unbound` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 88 unbound &&
useradd -c "Unbound DNS Resolver" -d /var/lib/unbound -u 88 \
-g unbound -s /bin/false unbound
```

Install Unbound by running the following commands:

```
./configure --prefix=/usr      \
--sysconfdir=/etc \
--disable-static \
--with-pidfile=/run/unbound.pid &&
make
```

If you have [Doxygen-1.12.0](#) package installed and want to build html documentation, run the following command:

```
make doc
```

To test the results, issue `make check`.

Now, as the `root` user:

```
make install &&
mv -v /usr/sbin/unbound-host /usr/bin/
```

If you built the documentation, install it by running the following commands as the `root` user:

```
install -v -m755 -d /usr/share/doc/unbound-1.21.0 &&
install -v -m644 doc/html/* /usr/share/doc/unbound-1.21.0
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-libevent`: This option enables libevent support allowing use of large outgoing port ranges.

`--with-pyunbound`: This option enables building of the Python bindings.

Configuring Unbound

Config Files

`/etc/unbound/unbound.conf`

Configuration Information

In the default configuration, `unbound` will bind to localhost (127.0.0.1 IP address) and allow recursive queries only from localhost clients. If you want to use `unbound` for local DNS resolution, run the following command as the `root` user:

```
echo "nameserver 127.0.0.1" > /etc/resolv.conf
```

For advanced configuration see `/etc/unbound/unbound.conf` file and the documentation.

When Unbound is installed, some package builds fail if the file `/etc/unbound/root.key` is not found. Create this file by running the following command as the `root` user:

```
unbound-anchor
```

Systemd Unit

If you want the Unbound server to start automatically when the system is booted, install the `unbound.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-unbound
```

Contents

Installed Programs: `unbound`, `unbound-anchor`, `unbound-checkconf`, `unbound-control`, `unbound-control-setup`, and `unbound-host`

Installed Library: `libunbound.so`

Installed Directories: `/etc/unbound` and `/usr/share/doc/unbound-1.21.0` (optional)

Short Descriptions

<code>unbound</code>	is a DNS resolver daemon
<code>unbound-anchor</code>	performs setup or update of the root trust anchor for DNSSEC validation
<code>unbound-checkconf</code>	checks the <code>unbound</code> configuration file for syntax and other errors
<code>unbound-control</code>	performs remote administration on the <code>unbound</code> DNS resolver
<code>unbound-control-setup</code>	generates a self-signed certificate and private keys for the server and client
<code>unbound-host</code>	is a DNS lookup utility similar to <code>host</code> from BIND Utilities-9.20.0
<code>libunbound.so</code>	provides the Unbound API functions to programs

Part VI. Graphical Components

Chapter 24. Graphical Environments

This chapter contains instructions to build and configure a graphical user environment.

For a long time the only graphical environment usable with GNU/Linux has been the *X Window System*. It uses a client/server model which allows writing applications completely independent of the graphical hardware. This has the drawback that accessing modern hardware acceleration is difficult, so another approach named *Wayland* is developed. It is a simpler replacement for X, easier to develop and maintain, using the OpenGL framework. The main desktop environments GNOME and KDE have been ported to it.

This chapter provides the basic components of the X Window System and Wayland. For X, the chosen implementation is Xorg, which is a modular implementation and requires more than 100 packages to be installed. The distribution of Xorg is given a release number by the developers, in this case Xorg-7. Individual packages are updated as needed without changing this number.

Introduction to Xorg-7

Xorg is a freely redistributable, open-source implementation of the X Window System. This system provides a client/server interface between display hardware (the mouse, keyboard, and video displays) and the desktop environment, while also providing both the windowing infrastructure and a standardized application interface (API).

Xorg Download and Installation Instructions

Xorg-7.0 introduced a completely auto-tooled, modular build system. With the new modular build system, it is no longer possible to download the entire package in a single file. In fact, there will be well over 100 packages that need to be fetched from the download location. To assist with such a large task, installing [Wget-1.24.5](#) is strongly recommended for downloading the needed files. A complete wget file list is provided for each page that includes multiple packages.

Given the number of packages available, deciding which packages you need to install for your particular setup may seem a bit overwhelming at first. Take a look at [this page](#) and [this thread](#) to get an idea of what you will need. If you are unsure, you should install all packages at the cost of extra disk space.

Note

Even if you intend to download only the necessary packages, you should download the wget file lists. The list of files are ordered by dependency, and the package versions listed in the files are known to work well with each other. Further, the wget file lists contain comments for specific packages that are deprecated or are not recommended to install. Newer packages are likely intended for the next release of Xorg and have already proved to be incompatible with current versions of software installed in BLFS. The installed size of Xorg can be reduced considerably by installing only the packages that you will need and use, however, the BLFS book cannot account for all dependencies and build options for the individual Xorg packages. The instructions assume that all packages have been built.

Additionally, because of the large number of repetitive commands, you are encouraged to partially automate the build. Instructions have been given that utilize the [Sudo-1.9.15p5](#) package. It is recommended that you use the `:NOPASSWD` configuration option for the user that will be building the xorg packages.

Setting up the Xorg Build Environment

Note

The following instructions assume that the shell startup files have been set up as described in [The Bash Shell Startup Files](#).

As with previous releases of the X Window System, it may be desirable to install Xorg into an alternate prefix. This is no longer common practice among Linux distributions. The common installation prefix for Xorg on Linux is `/usr`. There is no standard alternate prefix, nor is there any exception in the current revision of the Filesystem Hierarchy Standard for Release 7 of the X Window System. Alan Coopersmith of Sun Microsystems, once stated "At Sun, we were using `/usr/X11` and plan to stick with it." Only the `/opt/*` prefix or the `/usr` prefix adhere to the current FHS guidelines.

The BLFS editors recommend using the `/usr` prefix.

Choose your installation prefix, and set the `XORG_PREFIX` variable with the following command:

```
export XORG_PREFIX=<PREFIX>
```

Throughout these instructions, you will use the following `configure` switches for all of the packages. Create the `XORG_CONFIG` variable to use for this parameter substitution:

```
export XORG_CONFIG="--prefix=$XORG_PREFIX --sysconfdir=/etc \
--localstatedir=/var --disable-static"
```

Note

We will use `$XORG_CONFIG` in the instructions for many packages belonging to or related to Xorg. These instructions won't work properly with the default behavior of [zsh-5.9](#). So if you are using [zsh-5.9](#) as the interactive shell and building a package for which the instruction uses this variable, make [zsh-5.9](#) behavior expanding `$XORG_CONFIG` same as bash:

```
set -o shwordsplit
```

If you want to make this setting persistent, add this command into the zsh startup file.

Create an `/etc/profile.d/xorg.sh` configuration file containing these variables as the `root` user:

```
cat > /etc/profile.d/xorg.sh << EOF
XORG_PREFIX="$XORG_PREFIX"
XORG_CONFIG="--prefix=\$XORG_PREFIX --sysconfdir=/etc --localstatedir=/var --disable-static"
export XORG_PREFIX XORG_CONFIG
EOF
chmod 644 /etc/profile.d/xorg.sh
```

Note

There is some confusion about the above 'here' document. The backslash in front of the dollar sign is correct. Bash will remove it when creating /etc/profile.d/xorg.sh. However, if you are creating the file with an editor, a copy and paste operation will not remove the backslash. It must then be removed manually.

If you've installed [Sudo-1.9.15p5](#), ensure that `XORG_PREFIX` and `XORG_CONFIG` are available in the sudo environment. As the `root` user, run the following command:

```
cat > /etc/sudoers.d/xorg << EOF
Defaults env_keep += XORG_PREFIX
Defaults env_keep += XORG_CONFIG
EOF
```

If you are not using the standard Xorg prefix...

Warning

If you've decided to use the standard `/usr` prefix, you must omit the remainder of this page and continue at [util-macros-1.20.1](#).

If you've decided to *not* use the standard prefix, be sure to add `$XORG_PREFIX/bin` to your `PATH` environment variable, and `$XORG_PREFIX/lib/pkgconfig` and `$XORG_PREFIX/share/pkgconfig` to your `PKG_CONFIG_PATH` variable. It is also helpful to specify additional search paths for `gcc` and an include directory for the `aclocal` program. Issue the following commands as the `root` user:

```
cat >> /etc/profile.d/xorg.sh << "EOF"
pathappend $XORG_PREFIX/bin          PATH
pathappend $XORG_PREFIX/lib/pkgconfig PKG_CONFIG_PATH
pathappend $XORG_PREFIX/share/pkgconfig PKG_CONFIG_PATH

pathappend $XORG_PREFIX/lib           LIBRARY_PATH
pathappend $XORG_PREFIX/include      C_INCLUDE_PATH
pathappend $XORG_PREFIX/include      CPLUS_INCLUDE_PATH

ACLOCAL="aclocal -I $XORG_PREFIX/share/aclocal"

export PATH PKG_CONFIG_PATH ACLOCAL LIBRARY_PATH C_INCLUDE_PATH CPLUS_INCLUDE_PATH
EOF
```

The script above needs to be activated. Normally it will be automatic at login, but to activate it now, as a regular user, run:

```
source /etc/profile.d/xorg.sh
```

You should also add `$XORG_PREFIX/lib` to the `/etc/ld.so.conf` file. Again, as the `root` user, issue the following command:

```
echo "$XORG_PREFIX/lib" >> /etc/ld.so.conf
```

You should also modify `/etc/man_db.conf`, adding appropriate `MANDATORY_MANPATH`, `MANPATH_MAP`, and `MANDB_MAP` entries following the examples for `/usr/X11R6`. Issue the following command as the `root` user:

```
sed -e "s@X11R6@man@X11R6/share/man@g" \
-e "s@/usr/X11R6@$XORG_PREFIX@g" \
-i /etc/man_db.conf
```

Some applications look for shared files in `/usr/share/X11`. Create a symbolic link to the proper location as the `root` user:

```
ln -svf $XORG_PREFIX/share/X11 /usr/share/X11
```

If building KDE, some cmake files look for Xorg in places other than `$XORG_PREFIX`. Allow cmake to find Xorg with:

```
ln -svf $XORG_PREFIX /usr/X11R6
```

util-macros-1.20.1

Introduction to util-macros

The util-macros package contains the m4 macros used by all of the Xorg packages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/util/util-macros-1.20.1.tar.xz>
- Download MD5 sum: 35a4f264a9852be4ae66f07d4100356e
- Download size: 84 KB
- Estimated disk space required: 520 KB
- Estimated build time: less than 0.1 SBU

util-macros Dependencies

Required

[Xorg build environment](#) (should be set for the following instructions to work)

Installation of util-macros

Install util-macros by running the following commands:

```
./configure $XORG_CONFIG
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: `$XORG_PREFIX/share/pkgconfig` and `$XORG_PREFIX/share/util-macros`

xorgproto-2024.1

Introduction to xorgproto

The xorgproto package provides the header files required to build the X Window system, and to allow other applications to build against the installed X Window system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xorg.freedesktop.org/archive/individual/proto/xorgproto-2024.1.tar.xz>
- Download MD5 sum: 12374d29fb5ae642cfa872035e401640
- Download size: 744 KB
- Estimated disk space required: 8.4 MB
- Estimated build time: less than 0.1 SBU

xorgproto Dependencies

Required

[util-macros-1.20.1](#)

Optional

[fop-2.9](#), [libxslt-1.1.42](#), [xmlto-0.0.29](#) and [asciidoc-10.2.1](#) (to build additional documentation)

Note

There is a reciprocal dependency with [fop-2.9](#). If you wish to build the documentation, you'll need to re-install the Protocol Headers after the installation is complete and [fop-2.9](#) has been installed.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/Xorg7ProtocolHeaders>

Installation of xorgproto

Install xorgproto by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=$XORG_PREFIX .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install &&
mv -v $XORG_PREFIX/share/doc/xorgproto{-2024.1}
```

Command Explanations

`-D legacy=true`: Installs legacy headers needed by old programs (such as [LessTif](#)).

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: `$XORG_PREFIX/include/GL`, `$XORG_PREFIX/include/X11`, and `$XORG_PREFIX/share/doc/xorgproto-2024.1`

libXau-1.0.11

Introduction to libXau

The libXau package contains a library implementing the X11 Authorization Protocol. This is useful for restricting client access to the display.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/lib/libXau-1.0.11.tar.xz>
- Download MD5 sum: 7f14ba9c84a81a2b9dd023706febab38
- Download size: 268 KB
- Estimated disk space required: 2.8 MB (with test)
- Estimated build time: less than 0.1 SBU (with test)

libXau Dependencies

Required

[xorgproto-2024.1](#)

Installation of libXau

Install libXau by running the following commands:

```
./configure $XORG_CONFIG &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libXau.so

Installed Directories: None

Short Descriptions

`libXau.so` is the library of X authority database routines

libXdmcp-1.1.5

Introduction to libXdmcp

The libXdmcp package contains a library implementing the X Display Manager Control Protocol. This is useful for allowing clients to interact with the X Display Manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/lib/libXdmcp-1.1.5.tar.xz>
- Download MD5 sum: ce0af51de211e4c99a111e64ae1df290
- Download size: 292 KB
- Estimated disk space required: 3.0 MB (with test)
- Estimated build time: less than 0.1 SBU (with test)

libXdmcp Dependencies

Required

[xorgproto-2024.1](#)

Optional

[xmlto-0.0.29](#), [fop-2.9](#), [libxslt-1.1.42](#), and [Xorg-SGML-doctools](#) (for documentation)

Installation of libXdmcp

Install libXdmcp by running the following commands:

```
./configure $XORG_CONFIG --docdir=/usr/share/doc/libXdmcp-1.1.5 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libXdmcp.so

Installed Directory: \$XORG_PREFIX/share/doc/libXdmcp-1.1.5

Short Descriptions

libXdmcp.so is the X Display Manager Control Protocol library

xcb-proto-1.17.0

Introduction to xcb-proto

The xcb-proto package provides the XML-XCB protocol descriptions that libxcb uses to generate the majority of its code and API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xorg.freedesktop.org/archive/individual/proto/xcb-proto-1.17.0.tar.xz>
- Download MD5 sum: c415553d2ee1a8cea43c3234a079b53f
- Download size: 152 KB
- Estimated disk space required: 1.3 MB
- Estimated build time: less than 0.1 SBU

xcb-proto Dependencies

Recommended

[Xorg build environment](#) (needed for the instructions below)

Optional

[libxml2-2.13.3](#) (required to run the tests)

Installation of xcb-proto

Install xcb-proto by running the following commands:

```
PYTHON=python3 ./configure $XORG_CONFIG
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you are upgrading from version 1.15.1 or lower, the old pkgconfig file needs to be removed. Issue, as the `root` user:

```
rm -f $XORG_PREFIX/lib/pkgconfig/xcb-proto.pc
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: `$XORG_PREFIX/share/xcb` and `$XORG_PREFIX/lib/python3.12/site-packages/xcbgen`

libxcb-1.17.0

Introduction to libxcb

The libxcb package provides an interface to the X Window System protocol, which replaces the current Xlib interface. Xlib can also use XCB as a transport layer, allowing software to make requests and receive responses with both.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xorg.freedesktop.org/archive/individual/lib/libxcb-1.17.0.tar.xz>
- Download MD5 sum: 96565523e9f9b701fcb35d31f1d4086e
- Download size: 448 KB
- Estimated disk space required: 30 MB (with tests, add 62 MB for doxygen docs)
- Estimated build time: 0.2 SBU (with tests, add 1.4 SBU for doxygen docs)

libxcb Dependencies

Required

[libXau-1.0.11](#) and [xcb-proto-1.17.0](#)

Recommended

[libXdmcp-1.1.5](#)

Optional

[Doxygen-1.12.0](#) (to generate API documentation) and [libxslt-1.1.42](#)

Installation of libxcb

Install libxcb by running the following commands:

```
./configure $XORG_CONFIG \
--without-doxygen \
--docdir='${datadir}'/doc/libxcb-1.17.0 &&
LC_ALL=en_US.UTF-8 make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If the package was built as a non-`root` user, the installed documentation is now owned by this user. As the `root` user, fix the ownership:

```
chown -R root:root $XORG_PREFIX/share/doc/libxcb-1.17.0
```

Command Explanations

`LC_ALL=en_US.UTF-8`: Some Python 3 script in the building system of this package may fail with certain system locale settings. This environment variable forces a locale setting known to work.

`--without-doxygen`: Do not use doxygen to generate API documentation (default: auto). Without it, if [Doxygen-1.12.0](#) is installed, the API documentation will be generated and installed.

Contents

Installed Programs: None

Installed Libraries: libxcb.so, libxcb-composite.so, libxcb-damage.so, libxcb-dbe.so, libxcb-dpms.so, libxcb-dri2.so, libxcb-dri3.so, libxcb-glx.so, libxcb-present.so, libxcb-randr.so, libxcb-record.so, libxcb-render.so, libxcb-res.so, libxcb-screensaver.so, libxcb-shape.so, libxcb-shm.so, libxcb-sync.so, libxcb-xf86dri.so, libxcb-xfixes.so, libxcb-xinerama.so, libxcb-xinput.so, libxcb-xkb.so, libxcb-xtest.so, libxcb-xvmc.so, and libxcb-xv.so

Installed Directories: `$XORG_PREFIX/include/xcb` and `$XORG_PREFIX/share/doc/libxcb-1.17.0`

Short Descriptions

`libxcb.so` is an interface to the X Window System protocol

Introduction to Xorg Libraries

The Xorg libraries provide library routines that are used within all X Window applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/lib/>
- Download size: 12 MB
- Estimated disk space required: 225 MB (34 MB installed) - if source directories not deleted
- Estimated build time: 2.1 SBU

Xorg Libraries Dependencies

Required

[Fontconfig-2.15.0](#) and [libxcb-1.17.0](#)

Optional

[asciidoc-10.2.1](#), [xmlto-0.0.29](#) with one or more of the following: [fop-2.9](#), [Links-2.30](#), [Lynx-2.9.2](#), [ncompress](#) (for some tests), and [W3m](#) (to generate additional PDF or text documentation for the libXfont package).

Recommended at runtime

[dbus-1.14.10](#)

Downloading Xorg Libraries

First, create a list of files to be downloaded. This file will also be used to verify the integrity of the downloads when complete:

```
cat > lib-7.md5 << "EOF"
12344cd74a1eb25436ca6e6a2cf93097  xtrans-1.5.0.tar.xz
5b8fa54e0ef94136b56f887a5e6cf6c9  libX11-1.8.10.tar.xz
e59476db179e48c1fb4487c12d0105d1  libXext-1.3.6.tar.xz
c5cc0942ed39c49b8fcfd47a427bd4305  libFS-1.0.10.tar.xz
b444a0e4c2163d1bbc7b046c3653eb8d  libICE-1.1.1.tar.xz
ffa434ed96ccae45533b3d653300730e  libSM-1.2.4.tar.xz
e613751d38e13aa0d0fd8e0149ec057  libXScrnSaver-1.2.4.tar.xz
4ea21d3b5a36d93a2177d9abed2e54d4  libXt-1.3.0.tar.xz
85edefb7deaad4590a03fccba517669f  libXmu-1.2.1.tar.xz
05b5667aadd476d77e9b5balade213e  libXpm-3.5.17.tar.xz
2a9793533224f92ddad256492265dd82  libXaw-1.0.16.tar.xz
65b9ba1e9ff3d16c4fa72915d4bb585a  libXfixes-6.0.1.tar.xz
af0a5f0abb5b55f8411cd738cf0e5259  libXcomposite-0.4.6.tar.xz
ebf7fb3241ec03e8a3b2af72f03b4631  libXrender-0.9.11.tar.xz
bf3a43ad8cb91a258b48f19c83af8790  libXcursor-1.2.2.tar.xz
ca55d29fa0a8b5c4a89f609a7952ebf8  libXdamage-1.1.6.tar.xz
8816cc44d06ebe42e85950b368185826  libfontenc-1.1.8.tar.xz
66e03e3405d923dfaf319d6f2b47e3da  libXfont2-2.0.7.tar.xz
cea0a3304e47a841c90fbeeeb55329ee  libXft-2.3.8.tar.xz
89ac74ad6829c08d5c8ae8f48d363b06  libXi-1.8.1.tar.xz
228c877558c265d2f63c56a03f7d3f21  libXinerama-1.1.5.tar.xz
24e0b72abe16efce9bf10579beaffc27  libXrandr-1.5.4.tar.xz
66c9e9e01b0b53052bb1d02ebf8d7040  libXres-1.2.2.tar.xz
b62dc44d8e63a67bb10230d54c44dcb7  libXst-1.2.5.tar.xz
70bfd14ca1a563c218794413fc01f42  libXv-1.0.12.tar.xz
a90a5f01102dc445c7decbbd9ef77608  libXvMC-1.0.14.tar.xz
74dlacf93b83abeb0954824da0ec400b  libXxf86dga-1.1.6.tar.xz
5b913dac587f2de17a02e17f9a44a75f  libXxf86vm-1.1.5.tar.xz
57c7efbeceedefde006123a77a7bc825  libpciaccess-0.18.1.tar.xz
229708c15c9937b6e5131d0413474139  libxkbfile-1.1.3.tar.xz
faa74f7483074ce7d4349e6bdc237497  libxshmfence-1.3.2.tar.xz
bdd3ec17c6181fd7b26f6775886c730d  libXpresent-1.0.1.tar.xz
EOF
```

To download the needed files using [Wget-1.24.5](#), use the following commands:

```

mkdir lib &&
cd lib &&
grep -v '^#' ../lib-7.md5 | awk '{print $2}' | wget -i- -c \
-B https://www.x.org/pub/individual/lib/ &&
md5sum -c ../lib-7.md5

```

Installation of Xorg Libraries

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

1. Run the entire script as the root user (not recommended).
2. Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
3. Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```

as_root()
{
    if [ $EUID = 0 ]; then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\"\\"
    fi
}

export -f as_root

```

Some libraries come with a test suite. If you wish to execute them, either comment out the `rm -rf ...` below, so that, after all libraries are installed, you can come back to the corresponding directory and run `make check`, or do individual builds, running the tests for each of those distributed with working test suites. Alternatively, you can uncomment the line `#make check ...`, and at the end, check the test results with:

```
grep -A9 summary *make_check.log
```

BLFS developers have confirmed that libX11, libXt, libXmu, libXpm, and libxshm fence are distributed with working test suites.

First, start a subshell that will exit on error:

```
bash -e
```

Install all of the packages by running the following commands:

```

for package in $(grep -v '^#' ../lib-7.md5 | awk '{print $2}')
do
    packagedir=${package%.tar.*}
    echo "Building $packagedir"

    tar -xf $package
    pushd $packagedir
    docdir="--docdir=$XORG_PREFIX/share/doc/$packagedir"

    case $packagedir in
        libXfont2-[0-9]* )
            ./configure $XORG_CONFIG $docdir --disable-devel-docs
        ;;

        libXt-[0-9]* )
            ./configure $XORG_CONFIG $docdir \
                --with-appdefaultdir=/etc/X11/app-defaults
        ;;

        libXpm-[0-9]* )
            ./configure $XORG_CONFIG $docdir --disable-open-zfile
        ;;
    esac
done

```

```

libpciaccess* )
mkdir build
cd build
meson setup --prefix=$XORG_PREFIX --buildtype=release ..
ninja
#ninja test
as_root ninja install
popd # $packagedir
continue # for loop
;;

* )
./configure $XORG_CONFIG $docdir
;;
esac

make
#make check 2>&1 | tee ../$packagedir-make_check.log
as_root make install
popd
rm -rf $packagedir
as_root /sbin/ldconfig
done

```

Finally, exit the shell that was started earlier:

```
exit
```

Command Explanations

--disable-open-zfile: Allow libXpm to build without the optional `compress` command present.

--disable-devel-docs: Disable generation of text documentation in the libXfont2 package if [xmlto-0.0.29](#) is installed without a text browser. Omit this parameter (or the entire `case` statement) if a text browser is installed.

--with-fop: Use [fop-2.9](#) to generate PDF documentation (only for the libXfont package).

Configuration of Xorg Libraries

If you've chosen to install Xorg into `/usr`, then no further configuration is necessary and you can skip the rest of this section. If you've opted for an alternate prefix, you should create two symlinks to satisfy the expected environment of several packages. Execute the following commands as the root user:

```
ln -sv $XORG_PREFIX/lib/X11 /usr/lib/X11 &&
ln -sv $XORG_PREFIX/include/X11 /usr/include/X11
```

Contents

Installed Programs: cxpm and sxpm

Installed Libraries: libfontenc.so, libFS.so, libICE.so, libpciaccess.so, libSM.so, libX11.so, libX11-xcb, libXaw6.so, libXaw7.so, libXaw.so, libXcomposite.so, libXcursor.so, libXdamage.so, libXext.so, libXfixes.so, libXfont2.so, libXft.so, libXinerama.so, libXi.so, libXkbfile.so, libXmu.so, libXmuu.so, libXpm.so, libXpresent.so, libXrandr.so, libXrender.so, libXRes.so, libXshmfence.so, libXss.so, libXt.so, libXtst.so, libXvMC.so, libXvMCW.so, libXv.so, libXxf86dga.so and libXxf86vm.so

Installed Directories: \$XORG_PREFIX/include/X11/fonts, \$XORG_PREFIX/include/X11/ICE, \$XORG_PREFIX/include/X11/SM, \$XORG_PREFIX/include/X11/Xmu, \$XORG_PREFIX/include/X11/Xtrans, \$XORG_PREFIX/share/doc/libFS, \$XORG_PREFIX/share/doc/libICE-1.1.1, \$XORG_PREFIX/share/doc/libSM-1.2.4, \$XORG_PREFIX/share/doc/libX11-1.8.10, \$XORG_PREFIX/share/doc/libXaw, \$XORG_PREFIX/share/doc/libXext, \$XORG_PREFIX/share/doc/libXi, \$XORG_PREFIX/share/doc/libXmu-1.2.1, \$XORG_PREFIX/share/doc/libXrender, \$XORG_PREFIX/share/doc/libXt, \$XORG_PREFIX/share/doc/libXtst, \$XORG_PREFIX/share/doc/libXvMC, \$XORG_PREFIX/share/doc/xtrans and \$XORG_PREFIX/share/X11/locale

Short Descriptions

<code>cxpm</code>	checks the format of an XPM file
<code>sxpm</code>	shows an XPM file and/or converts XPM 1 or 2 files to XPM 3
<code>libfontenc.so</code>	is the X11 font encoding library
<code>libFS.so</code>	is the library interface to the X Font Server

libICE.so	is the X Inter Client Exchange Library
libpciaccess.so	is the generic PCI Access library for X
libSM.so	is the X Session Management Library
libX11.so	is the Xlib Library
libXaw6.so	is the X Athena Widgets Library, version 6
libXaw7.so	is the X Athena Widgets Library, version 7
libXaw.so	are symbolic links to the current X Athena Widgets Library, version 7
libXcomposite.so	is the X Composite Library
libXcursor.so	is the X Cursor management library
libXdamage.so	is the X Damage Library
libXext.so	is the Misc X Extension Library
libXfixes.so	provides augmented versions of core protocol requests
libXfont2.so	is the X font library
libXft.so	is the X FreeType interface library
libXinerama.so	is the Xinerama Library
libXi.so	is the X Input Extension Library
libxkbfile.so	is the xkbfile Library
libXmu.so	is the X interface library for miscellaneous utilities not part of the Xlib standard
libXmuu.so	is the Mini Xmux Library
libXpm.so	is the X Pixmap Library
libXpresent.so	is the library interface to the X Present Extension
libXrandr.so	is the X Resize, Rotate and Reflection extension library
libXrender.so	is the X Render Library
libXRes.so	is the X-Resource extension client library
libxshmfence.so	exposes an event API on top of Linux futexes
libXss.so	is the X11 Screen Saver extension client library
libXt.so	is the X Toolkit Library
libXtst.so	is the Xtst Library
libXvMC.so	is the X-Video Motion Compensation Library
libXvMCW.so	is the XvMC Wrapper including the Nonstandard VLD extension
libXv.so	is the X Window System video extension library
libXxf86dga.so	is the client library for the XFree86-DGA extension
libXxf86vm.so	is the client library for the XFree86-VidMode X extension

libxcvt-0.1.2

Introduction to libxcvt

libxcvt is a library providing a standalone version of the X server implementation of the VESA CVT standard timing modelines generator. It is meant to be a direct replacement to the version formerly provided by the Xorg server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/lib/libxcvt-0.1.2.tar.xz>
- Download MD5 sum: b553fdb6024c5a137ff925bf4c337724
- Download size: 12 KB
- Estimated disk space required: 476 KB
- Estimated build time: less than 0.1 SBU

libxcvt Dependencies

Required

[Xorg build environment](#) (should be set for the following instructions to work)

Installation of libxcvt

Install libxcvt by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=$XORG_PREFIX --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: cvt

Installed Library: libxcvt.so

Installed Directory: \$XORG_PREFIX/include/libxcvt

Short Descriptions

cvt	calculates VESA Coordinated Video Timing (CVT) modelines for use with X
libxcvt.so	contains functions for calculating VESA CVT

xcb-util-0.4.1

Introduction to xcb-util

The xcb-util package provides additional extensions to the XCB library, many that were previously found in Xlib, but are not part of core X protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xcb.freedesktop.org/dist/xcb-util-0.4.1.tar.xz>
- Download MD5 sum: 34d749eab0fd0ffd519ac64798d79847
- Download size: 261 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: less than 0.1 SBU

xcb-util Dependencies

Required

[libxcb-1.17.0](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of xcb-util

Install xcb-util by running the following commands:

```
./configure $XORG_CONFIG &&
```

```
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libxcb-util.so

Installed Directories: None

Short Descriptions

`libxcb-util.so` Provides utility functions for other XCB utilities

xcb-util-image-0.4.1

Introduction to xcb-util-image

The xcb-util-image package provides additional extensions to the XCB library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xcb.freedesktop.org/dist/xcb-util-image-0.4.1.tar.xz>
- Download MD5 sum: a67bfac2eff696170259ef1f5ce1b611
- Download size: 284 KB
- Estimated disk space required: 2.9 MB
- Estimated build time: less than 0.1 SBU

xcb-util-image Dependencies

Required

[xcb-util-0.4.1](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of xcb-util-image

Install xcb-util-image by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

To test the results, issue: `LD_LIBRARY_PATH=$XORG_PREFIX/lib make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libxcb-image.so

Installed Directories: None

Short Descriptions

xcb-util-keysyms-0.4.1

Introduction to xcb-util-keysyms

The xcb-util-keysyms package contains a library for handling standard X key constants and conversion to/from keycodes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xcb.freedesktop.org/dist/xcb-util-keysyms-0.4.1.tar.xz>
- Download MD5 sum: fbdc05f86f72f287ed71b162f1a9725a
- Download size: 256 KB
- Estimated disk space required: 2.3 MB
- Estimated build time: less than 0.1 SBU

xcb-util-keysyms Dependencies

Required

[libxcb-1.17.0](#)

Optional

[Doxygen-1.12.0](#) (to generate documentation)

Installation of xcb-util-keysyms

Install xcb-util-keysyms by running the following commands:

```
./configure $XORG_CONFIG &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libxcb-keysyms.so

Installed Directories: None

Short Descriptions

`libxcb-keysyms.so` provides the standard X key constants and API functions for conversion to/from keycodes

xcb-util-renderutil-0.3.10

Introduction to xcb-util-renderutil

The xcb-util-renderutil package provides additional extensions to the XCB library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xcb.freedesktop.org/dist/xcb-util-renderutil-0.3.10.tar.xz>
- Download MD5 sum: 193b890e2a89a53c31e2ece3afcbd55f

- Download size: 256 KB
- Estimated disk space required: 2.4 MB
- Estimated build time: less than 0.1 SBU

xcb-util-renderutil Dependencies

Required

[libxcb-1.17.0](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of xcb-util-renderutil

Install xcb-util-renderutil by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libxcb-render-util.so

Installed Directories: None

Short Descriptions

libxcb-render-util.so Provides convenience functions for the Render extension

xcb-util-wm-0.4.2

Introduction to xcb-util-wm

The xcb-util-wm package contains libraries which provide client and window-manager helpers for EWMH and ICCCM.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xcb.freedesktop.org/dist/xcb-util-wm-0.4.2.tar.xz>
- Download MD5 sum: 581b3a092e3c0c1b4de6416d90b969c3
- Download size: 280 KB
- Estimated disk space required: 3.2 MB
- Estimated build time: less than 0.1 SBU

xcb-util-wm Dependencies

Required

[libxcb-1.17.0](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of xcb-util-wm

Install xcb-util-wm by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libxcb-ewmh.so and libxcb-icccm.so

Installed Directories: None

Short Descriptions

libxcb-ewmh.so	provides the client and window-manager helpers for EWMH
libxcb-icccm.so	provides the client and window-manager helpers for ICCCM

xcb-util-cursor-0.1.4

Introduction to xcb-util-cursor

The xcb-util-cursor package provides a module that implements the XCB cursor library. It is the XCB replacement for libXcursor.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://xcb.freedesktop.org/dist/xcb-util-cursor-0.1.4.tar.xz>
- Download MD5 sum: 0d244518ad54b886413fe782235d6210
- Download size: 260 KB
- Estimated disk space required: 2.5 MB
- Estimated build time: less than 0.1 SBU

xcb-util-cursor Dependencies

Required

[xcb-util-image-0.4.1](#) and [xcb-util-renderutil-0.3.10](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of xcb-util-cursor

Install xcb-util-cursor by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libxcb-cursor.so

Installed Directories: None

Short Descriptions

libxcb-cursor.so Is a port of Xlib's libXcursor functions

Mesa-24.1.5

Introduction to Mesa

Mesa is an OpenGL compatible 3D graphics library.

Note

Mesa is updated relatively often. You may want to use the latest available 24.1.x mesa version.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://mesa.freedesktop.org/archive/mesa-24.1.5.tar.xz>
- Download MD5 sum: 0f4de2192e63b438637b98937ac85a10
- Download size: 28 MB
- Estimated disk space required: 825 MB (with docs, add 496 MB for tests)
- Estimated build time: 4.1 SBU (with docs; add 1.4 SBU for tests; both with parallelism=4)

Additional Downloads

- Recommended patch: https://www.linuxfromscratch.org/patches/blfs/12.2/mesa-add_xdemos-2.patch
(installs 2 demo programs for testing Mesa - not needed if you install the [mesa-demos](#) package)

Mesa Dependencies

Required

[Xorg Libraries](#), [libdrm-2.4.122](#), and [Mako-1.3.5](#)

Recommended

- [Glslang-14.3.0](#) (required for Vulkan support)
- [libva-2.22.0](#) (to provide VA-API support for some gallium drivers. Note that there is a circular dependency. You must build libva first without mesa's EGL and GLX support, install this package, and rebuild libva)
- [libvdpau-1.5](#) (to build VDPAU drivers)
- [LLVM-18.1.7](#) (required for the r300, r600, and radeonsi drivers, and the LLVMpipe software rasterizer which can make the swrast driver much faster)
- [wayland-protocols-1.36](#) (required for [Plasma-6.1.4](#), GNOME, and recommended for [GTK+-3.24.43](#))
- [libclc-18.1.7](#) (required for the Intel iris gallium driver)
- [Vulkan-Loader-1.3.294](#) (required for the Zink gallium driver)
- [ply-3.11](#) (required for the Intel vulkan driver)
- [Cbindgen-0.27.0](#) and [rust-bindgen-0.70.0](#) (required for the Nouveau Vulkan driver)

Note

Choosing Mesa Drivers

In the instructions below, all the available drivers are built. That will almost always work. However, it is not efficient. Depending on your video hardware, you probably need only specific drivers.

The first thing you need to know is which type of video device you have. In some cases it is built into the CPU. In others it is a separate PCI card. In either case you can tell what video hardware you have by installing [pciutils-3.13.0](#) and running:

```
lspci | grep VGA
```

The video device is most likely one of three families: AMD, Intel, or Nvidia. See the Command Explanations for [-D gallium-drivers=auto](#) below to see what options are available for your specific video hardware (or emulated video hardware). You should probably add the swrast option as a backup driver.

For 'platforms' you can select x11 and/or wayland. Note that in BLFS currently only Gnome and KDE can use wayland. If you are not going to use one of those desktop environments then you probably only want x11.

For 'vulkan-drivers' you may want to limit the selection to your current hardware. These drivers are used by some specific applications. For example, ffmpeg (including ffplay) will use Vulkan instead of OpenGL for rendering video on the GPU. If you do not want to do this you may or may not want to install them. See the Command Explanations for [-D vulkan-drivers=auto](#) below to see what options are available for your specific video hardware (or emulated video hardware). You should probably add the swrast option as a backup driver unless you don't need Vulkan at all.

Optional

[libgcrypt-1.11.0](#), [libunwind-1.6.2](#), [lm-sensors-3-6-0](#) , [Nettle-3.10](#), [Valgrind-3.23.0](#), [mesa-demos](#) (provides more than 300 extra demos to test Mesa; this includes the same programs added by the patch above), [Bellagio OpenMAX Integration Layer](#) (for mobile platforms), and [libtizonia](#),

Note

An Internet connection is needed for building this package.

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
Device Drivers --->
Graphics support --->
  <*/M> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
    ...
    [DRM]
    # For r300 or r600:
    < */M> ATI Radeon
    ...
    [DRM_RADEON]
    # For radeonsi:
    < */M> AMD GPU
    [*]   Enable amdgpu support for SI parts
    [*]   Enable amdgpu support for CIK parts
    ...
    [DRM_AMDGPU_SI]
    [DRM_AMDGPU_CIK]
    Display Engine Configuration --->
    [*]   AMD DC - Enable new display engine
    ...
    [DRM_AMD_DC]
    # For nouveau:
    < */M> Nouveau (NVIDIA) cards
    ...
    [DRM_NOUVEAU]
    # For i915, crocus, or iris:
    < */M> Intel 8xx/9xx/G3x/G4x/HD Graphics
    ...
    [DRM_I915]
    # For swrast:
    < */M> Virtual GEM provider
    ...
    [DRM_VGEM]
    # For svga:
    < */M> DRM driver for VMware Virtual GPU
    ...
    [DRM_VMWGFX]
```

Note

The corresponding Mesa Gallium3D driver name is provided as the comment for the configuration entries. If you don't know the name of the Mesa Gallium3D driver for your GPU, see [Mesa Gallium3D Drivers](#) below.

`CONFIG_DRM_RADEON`, `CONFIG_DRM_AMDGPU`, `CONFIG_DRM_NOUVEAU`, and `CONFIG_DRM_I915` may require firmware. See [About Firmware](#) for details.

Selecting `CONFIG_DRM_RADEON` or `CONFIG_DRM_AMDGPU` as "y" is not recommended. If it is, any required firmware must be built as a part of the kernel image or the initramfs for the driver to function correctly.

The sub-entries under `CONFIG_DRM_AMDGPU` are used to ensure the AMDGPU kernel driver supports all GPUs using the `radeon` driver. They are not needed if you won't need `CONFIG_DRM_AMDGPU` itself. They may be unneeded for some GPU models.

For `swrast`, `CONFIG_DRM_VGEM` is not strictly needed but recommended as an optimization.

Installation of Mesa

If you have downloaded the `xidemos` patch (needed if testing the Xorg installation per BLFS instructions), apply it by running the following command:

```
patch -Np1 -i ../mesa-add_xdemos-2.patch
```

Install Mesa by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=$XORG_PREFIX \
    --buildtype=release \
    -D platforms=x11,wayland \
    -D gallium-drivers=auto \
    -D vulkan-drivers=auto \
    -D valgrind=disabled \
    -D libunwind=disabled &&
ninja
```

To test the results, issue: `meson configure -D build-tests=true && ninja test`.

Now, as the `root` user:

```
ninja install
```

If desired, install the optional documentation by running the following commands as the `root` user:

```
cp -rv ../docs -T /usr/share/doc/mesa-24.1.5
```

Command Explanations

`--buildtype=release`: This switch ensures a fully-optimized build, and disables debug assertions which will severely slow down the libraries in certain use-cases. Without this switch, build sizes can span into the 2 GB range.

`-D gallium-drivers=auto`: This parameter controls which Gallium3D drivers should be built:

- `auto` selects all Gallium3D drivers available for x86. With a comma-separated list, only a subset of these drivers will be built. If you precisely know which drivers you need, you can select them explicitly. For example, `-D gallium-drivers=radeon,iris,swrast`.
- `r300` (for ATI Radeon 9000 or Radeon X series)
- `r600` (for AMD/ATI Radeon HD 2000-6000 series)
- `radeon` (for AMD Radeon HD 7000 or newer AMD GPU models)
- `nouveau` (for supported NVIDIA GPUs, they are listed as all "3D features" either "DONE" or "N/A" in [the Nouveau status page](#))
- `virgl` (for a QEMU virtual GPU with virglrender support; note that BLFS [gemu-9.0.2](#) is not built with virglrender)
- `svga` (for a VMWare virtual GPU)
- `swrast` (using CPU for 3D rasterization. Note that it's much slower than using a modern 3D-capable GPU, so it should be only used if the GPU is not supported by other drivers),
- `iris` (for Intel GPUs shipped with Broadwell or newer CPUs, or as a dedicated PCIe card)
- `crocus` (for Intel GMA 3000, X3000 series, 4000 series, or X4000 series GPUs shipped with chipsets, or Intel HD GPUs shipped with pre-Broadwell CPUs)
- `i915` (for Intel GMA 900, 950, 3100, or 3150 GPUs shipped with chipsets or Atom D/N 4xx/5xx CPUs)
- `zink` (using Vulkan to implement OpenGL, it's not very useful on x86 as at now but it may replace other drivers for modern

GPU models in the future)

`-D vulkan-drivers=auto`: This parameter controls which Vulkan drivers should be built:

- `auto` selects all Vulkan drivers available for x86. With a comma-separated list, only a subset of these drivers will be built. If you precisely know which drivers you need, you can select them explicitly. For example, `-D vulkan-drivers=amd,nouveau,swrast`.
- `amd` (for AMD Radeon HD 7730 or newer AMD GPUs)
- `intel` (for Intel GPUs shipped with Skylake or newer CPUs, or as a dedicated PCIe card)
- `intel_hasvk` (for Intel GPUs shipped with Ivy Bridge, Haswell, or Broadwell CPUs)
- `nouveau` (for GTX 16XX, RTX 20XX, or newer NVIDIA GPUs)
- `swrast` (for using the CPU for 3D rasterization) Note that it's much slower than using a modern 3D-capable GPU, so it should be only used if the GPU is not supported by other drivers.
- `""` (empty list, use `-D vulkan-drivers=""` if you've not installed and you don't plan to install [Vulkan-Loader-1.3.294](#))

`-D platforms=...:` This parameter controls which windowing systems will be supported. Available linux platforms are `x11` and `wayland`.

`-D valgrind=disabled`: This parameter disables the usage of Valgrind during the build process. Remove this parameter if you have Valgrind installed, and wish to check for memory leaks.

`-D libunwind=disabled`: This parameter disables the usage of libunwind.

`meson configure -D build-tests=true`: This command will reconfigure the build to set `-D build-tests=true`, but keep the other options specified in the `meson setup` command unchanged. It allows `ninja test` to build and run unit tests.

`-D egl-native-platform="...":` This parameter controls which Embedded Graphics Library support will be built. Available linux options are `auto` (default), `x11`, `wayland`, `surfaceless`, and `drm`.

Contents

Installed Programs: `glxgears`, `glxinfo`, `mme_fermi_sim_hw_test`, and `mme_tu104_sim_hw_test`

Installed Libraries: `libEGL.so`, `libGL.so`, `libGLESv1_CM.so`, `libGLESv2.so`, `libgbm.so`, `libglapi.so`, and `libxatracker.so`

Installed DRI Drivers: `crocus_dri.so`, `i915_dri.so`, `iris_dri.so`, `kms_swrast_dri.so`, `nouveau_dri.so`, `nouveau_drv_video.so`, `r300_dri.so`, `r600_dri.so`, `r600_drv_video.so`, `radeonsi_dri.so`, `radeonsi_drv_video.so`, `swrast_dri.so`, `virtio_gpu_dri.so`, `virtio_gpu_drv_video.so`, `vmwgfx_dri.so`, and `zink_dri.so`

Installed VDPAU Drivers: `libvdpau_nouveau.so`, `libvdpau_r600.so`, and `libvdpau_radeonsi.so` `libvdpau_virtio_gpu.so` (Many of these drivers are hard-linked).

Installed Vulkan Drivers: `libvulkan_intel_hasvk.so`, `libvulkan_intel.so`, `libvulkan_lvp.so`, `libvulkan_nouveau.so`, and `libvulkan_radeon.so`

Installed Directories: `$XORG_PREFIX/include/{EGL,GLES,GLES2,GLES3,KHR}`, `$XORG_PREFIX/lib/{dri,vdpau}`, `$XORG_PREFIX/share/drirc.d` (contains workarounds for various applications, particularly browsers and games), `$XORG_PREFIX/share/vulkan`, and `/usr/share/doc/mesa-24.1.5`

Short Descriptions

<code>glxgears</code>	is a GL demo useful for troubleshooting graphics problems
<code>glxinfo</code>	is a diagnostic program that displays information about the graphics hardware and installed GL libraries
<code>libEGL.so</code>	provides a native platform graphics interface as defined by the EGL-1.4 specification
<code>libgbm.so</code>	is the Mesa Graphics Buffer Manager library
<code>libGLESv1_CM.so</code>	is the Mesa OpenGL ES 1.1 library
<code>libGLESv2.so</code>	is the Mesa OpenGL ES 2.0 library
<code>libGL.so</code>	is the main Mesa OpenGL library

xbitmaps-1.1.3

Introduction to xbitmaps

The xbitmaps package contains bitmap images used by multiple applications built in Xorg chapter.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/data/xbitmaps-1.1.3.tar.xz>
- Download MD5 sum: 2b03f89d78fb91671370e77d7ad46907
- Download size: 108 KB
- Estimated disk space required: 1.6 MB
- Estimated build time: less than 0.1 SBU

xbitmaps Dependencies

Required

[util-macros-1.20.1](#)

Installation of xbitmaps

Install xbitmaps by running the following commands:

```
./configure $XORG_CONFIG
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: \$XORG_PREFIX/include/X11/bitmaps

Xorg Applications

Introduction to Xorg Applications

The Xorg applications provide the expected applications available in previous X Window implementations.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/app/>
- Download size: 4.9 MB
- Estimated disk space required: 48 MB
- Estimated build time: 1.5 SBU (ignoring the time to download)

Xorg Applications Dependencies

Required

[libpng-1.6.43](#), [Mesa-24.1.5](#), [xbitmaps-1.1.3](#), and [xcb-util-0.4.1](#)

Optional

[Linux-PAM-1.6.1](#) and both [cairo-5c](#) and [Nickle](#) (only if you wish to try to run the undocumented `xkeyhost` script).

Downloading Xorg Applications

First, create a list of files to be downloaded. This file will also be used to verify the integrity of the downloads when complete:

```

cat > app-7.md5 << "EOF"
30f898d71a7d8e817302970f1976198c iceauth-1.0.10.tar.xz
7dcf5f702781bddaaff02e963a56270 mkfontscale-1.2.3.tar.xz
05423bb42a006a6eb2c36ba10393de23 sessreg-1.1.3.tar.xz
1d61c9f4a3d1486eff575bf233e5776c setxkbmap-1.3.4.tar.xz
9f7a4305f0e79d5a46c3c7d02df9437d smproxy-1.0.7.tar.xz
e96b56756990c56c24d2d02c2964456b x11perf-1.6.1.tar.bz2
595c941d9aff6f6d6e038c4e42dcfff58 xauth-1.1.3.tar.xz
82a90e2feaaeb5c5e7610420930cc0f4 xcmsdb-1.0.6.tar.xz
89e81a1c31e4a1fb0e431425cd733d7 xcursorgen-1.0.8.tar.xz
933e6d65f96c890f8e96a9f21094f0de xdpyinfo-1.3.4.tar.xz
34aff1f93fa54d6a64cbe4fee079e077 xdriinfo-1.0.7.tar.xz
f29d1544f8dd126a1b85e2ff728672d xev-1.2.6.tar.xz
41afaa5a68cd0de7e7ece4805a37f11 xgamma-1.0.7.tar.xz
48ac13856838d34f2e7fca8cd1f1699 xhost-1.0.9.tar.xz
8e4d14823b7cbef1581c398c6ab0035 xinput-1.6.4.tar.xz
83d711948de9ccac550d2f4af50e94c3 xkbccomp-1.4.7.tar.xz
05ce1abd8533a400572784b1186a44d0 xkbevd-1.1.5.tar.xz
07483ddfe1d83c197df792650583ff20 xkbutils-1.0.6.tar.xz
f62b99839249ce9a7a8bb71a5bab6f9d xkill-1.0.6.tar.xz
da5b7a39702841281e1d86b7349a03ba xlsatoms-1.1.4.tar.xz
ab4b3c47e848ba8c3e47c021230ab23a xlsclients-1.1.5.tar.xz
ba2dd3db3361e374fefef2b1c797c46eb xmmessage-1.0.7.tar.xz
0d66e07595ae083871048c4b805d8b13 xmodmap-1.0.11.tar.xz
ab6c9d17eb1940acfcb80a72319270ae xpr-1.2.0.tar.xz
d050642a667b518cb3429273a59fa36d xprop-1.2.7.tar.xz
f822a8d5f233e609d27cc22d42a177cb xrandr-1.5.2.tar.xz
c8629d5a0bc878d10ac49elb290bf453 xrdb-1.2.2.tar.xz
55003733ef417db8fafce588ca74d584 xrefresh-1.1.0.tar.xz
18ff5cdff59015722431d568a5c0bad2 xset-1.2.5.tar.xz
fa9a24fe5b1725c52a4566a62dd0a50d xsetroot-1.1.3.tar.xz
d698862e9cad153c5fefca6eee964685 xvinfo-1.1.5.tar.xz
b0081fb92ae56510958024242ed1bc23 xwd-1.0.9.tar.xz
c91201bc1eb5e7b38933be8d0f7f16a8 xwininfo-1.1.6.tar.xz
5ff5dc120e8e927dc3c331c7fee33fc3 xwud-1.0.6.tar.xz
EOF

```

To download the needed files using [Wget-1.24.5](#), use the following commands:

```

mkdir app &&
cd app &&
grep -v '^#' ../app-7.md5 | awk '{print $2}' | wget -i - -c \
-B https://www.x.org/pub/individual/app/ &&
md5sum -c ../app-7.md5

```

Installation of Xorg Applications

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

1. Run the entire script as the root user (not recommended).
2. Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
3. Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```

as_root()
{
    if [ $EUID = 0 ];      then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\"\\"
    fi
}

export -f as_root

```

First, start a subshell that will exit on error:

```
bash -e
```

Install all of the packages by running the following commands:

```
for package in $(grep -v '^#' ../app-7.md5 | awk '{print $2}')
do
    packagedir=${package%.tar.?z*}
    tar -xf $package
    pushd $packagedir
    ./configure $XORG_CONFIG
    make
    as_root make install
    popd
    rm -rf $packagedir
done
```

Finally, exit the shell that was started earlier:

```
exit
```

Unless you installed the optional dependencies, remove an undocumented script which is reported to be broken (`xkeystone` provided by the `xrandr` package).

```
as_root rm -f $XORG_PREFIX/bin/xkeystone
```

Contents

Installed Programs: `iceauth`, `mkfontdir`, `mkfontscale`, `sessreg`, `setxkbmap`, `smproxy`, `x11perf`, `x11perfcomp`, `xauth`, `xcmsdb`, `xcursorgen`, `xdpr`, `xdpyinfo`, `xdriinfo`, `xev`, `xgamma`, `xhost`, `xinput`, `xkbbell`, `xkbcomp`, `xkbevd`, `xkbvleds`, `xkbwatch`, `xkill`, `xlsatoms`, `xlsclients`, `xmessage`, `xmodmap`, `xpr`, `xprop`, `xrandr`, `xrdb`, `xrefresh`, `xset`, `xsetroot`, `xvinfo`, `xwd`, `xwininfo`, and `xwud`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>iceauth</code>	is the ICE authority file utility
<code>mkfontdir</code>	creates an index of X font files in a directory
<code>mkfontscale</code>	creates an index of scalable font files for X
<code>sessreg</code>	manages utmp/wtmp entries for non-init clients
<code>setxkbmap</code>	sets the keyboard using the X Keyboard Extension
<code>smproxy</code>	is the Session Manager Proxy
<code>x11perf</code>	is an X11 server performance test program
<code>x11perfcomp</code>	is an X11 server performance comparison program
<code>xauth</code>	is the X authority file utility
<code>xcmsdb</code>	is the Device Color Characterization utility for the X Color Management System
<code>xcursorgen</code>	creates an X cursor file from a collection of PNG images
<code>xdpr</code>	dumps an X window directly to a printer
<code>xdpyinfo</code>	is a display information utility for X
<code>xdriinfo</code>	queries configuration information of DRI drivers
<code>xev</code>	prints contents of X events
<code>xgamma</code>	alters a monitor's gamma correction through the X server
<code>xhost</code>	is a server access control program for X
<code>xinput</code>	is a utility to configure and test X input devices
<code>xkbbell</code>	is an XKB utility program that raises a bell event
<code>xkbcomp</code>	compiles an XKB keyboard description
<code>xkbevd</code>	is the XKB event daemon
<code>xkbvleds</code>	shows the XKB status of keyboard LEDs
<code>xkbwatch</code>	monitors modifier keys and LEDs
<code>xkill</code>	kills a client by its X resource

<code>xlsatoms</code>	lists interned atoms defined on the server
<code>xlsclients</code>	lists client applications running on a display
<code>xmessage</code>	displays a message or query in a window
<code>xmodmap</code>	is a utility for modifying keymaps and pointer button mappings in X
<code>xpr</code>	prints an X window dump
<code>xprop</code>	is a property displayer for X
<code>xrandr</code>	is a primitive command line interface to RandR extension
<code>xrdb</code>	is the X server resource database utility
<code>xrefresh</code>	refreshes all or part of an X screen
<code>xset</code>	is the user preference utility for X
<code>xsetroot</code>	is the root window parameter setting utility for X
<code>xvinfo</code>	prints out X-Video extension adaptor information
<code>xwd</code>	dumps an image of an X window
<code>xwininfo</code>	is a window information utility for X
<code>xwud</code>	is an image displayer for X

luit-20240102

Introduction to luit

The luit package is a filter that can be run between an arbitrary application and a UTF-8 terminal emulator. It will convert application output from the locale's encoding into UTF-8, and convert terminal input from UTF-8 into the locale's encoding.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://invisible-mirror.net/archives/luit/luit-20240102.tgz>
- Download MD5 sum: b69a0819905ad8dc4bf82d25f24037a4
- Download size: 208 KB
- Estimated disk space required: 1.8 MB
- Estimated build time: less than 0.1 SBU

luit Dependencies

Required

[Xorg Applications](#)

Installation of luit

Install luit by running the following commands:

```
./configure $XORG_CONFIG &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: luit

Installed Libraries: None

Installed Directory: None

Introduction to xcursor-themes

The xcursor-themes package contains the redglass and whiteglass animated cursor themes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/data/xcursor-themes-1.0.7.tar.xz>
- Download MD5 sum: 070993be1f010b09447ea24bab2c9846
- Download size: 1.4 MB
- Estimated disk space required: 15 MB
- Estimated build time: less than 0.1 SBU

xcursor-themes Dependencies

Required

[Xorg Applications](#)

Installation of xcursor-themes

Note

We explicitly install the cursor themes in /usr instead of `$XORG_PREFIX` so non-Xorg desktop environments can find them.

Install xcursor-themes by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/icons/handhelds, /usr/share/icons/redglass, and /usr/share/icons/whiteglass

Xorg Fonts

Introduction to Xorg Fonts

The Xorg font packages provide some scalable fonts and supporting packages for Xorg applications. Many people will want to install other TTF or OTF fonts in addition to, or instead of, these. Some are listed at [the section called "TTF and OTF fonts".](#)

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/font/>
- Download size: 3.0 MB
- Estimated disk space required: 8.8 MB
- Estimated build time: 0.1 SBU

Xorg Fonts Dependencies

Required

[xcursor-themes-1.0.7](#)

Downloading Xorg Fonts

First, create a list of files to be downloaded. This file will also be used to verify the integrity of the downloads when complete:

```
cat > font-7.md5 << "EOF"
a6541d12ceba004c0c1e3df900324642  font-util-1.4.1.tar.xz
a56b1a7f2c14173f71f010225fa131f1  encodings-1.1.0.tar.xz
79f4c023e27d1db1fd90d041ce89835  font-alias-1.0.5.tar.xz
546d17feab30d4e3abcf332b454f58ed  font-adobe-utopia-type1-1.0.5.tar.xz
063bfa1456c8a68208bf96a33f472bb1  font-bh-ttf-1.0.4.tar.xz
51a17c981275439b85e15430a3d711ee  font-bh-type1-1.0.4.tar.xz
00f64a84b6c9886040241e081347a853  font-ibm-type1-1.0.4.tar.xz
fe972eaf13176fa9aa7e74a12ecc801a  font-misc-ethiopic-1.0.5.tar.xz
3b47fed2c032af3a32aad9acc1d25150  font-xfree86-type1-1.0.5.tar.xz
EOF
```

To download the needed files using [Wget-1.24.5](#), use the following commands:

```
mkdir font &&
cd font &&
grep -v '^#' ../font-7.md5 | awk '{print $2}' | wget -i - -c \
-B https://www.x.org/pub/individual/font/ &&
md5sum -c ../font-7.md5
```

Installation of Xorg Fonts

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

1. Run the entire script as the root user (not recommended).
2. Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
3. Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```
as_root()
{
    if [ $EUID = 0 ];      then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\\""
    fi
}

export -f as_root
```

First, start a subshell that will exit on error:

```
bash -e
```

Install all of the packages by running the following commands:

```
for package in $(grep -v '^#' ../font-7.md5 | awk '{print $2}')
do
    packagedir=${package%.tar.*}
    tar -xf $package
    pushd $packagedir
    ./configure $XORG_CONFIG
    make
    as_root make install
```

```
popd  
as_root rm -rf $packagedir  
done
```

Finally, exit the shell that was started earlier:

```
exit
```

When all of the fonts have been installed, the system must be configured so that Fontconfig can find the TrueType fonts. Since the fonts are outside of the default search path of several packages **if XORG_PREFIX is not /usr**, make symlinks to the Xorg TrueType font directories in `/usr/share/fonts` by running the following commands as the `root` user:

```
install -v -d -m755 /usr/share/fonts  
ln -svfn $XORG_PREFIX/share/fonts/X11/OTF /usr/share/fonts/X11-OTF &&  
ln -svfn $XORG_PREFIX/share/fonts/X11/TTF /usr/share/fonts/X11-TTF
```

Contents

Installed Programs: bdftruncate and ucs2any

Installed Libraries: None

Installed Directories: \$XORG_PREFIX/share/fonts

Short Descriptions

<code>bdftruncate</code>	generates a truncated BDF font from an ISO 10646-1 encoded BDF font
<code>ucs2any</code>	generates BDF fonts in any encoding from an ISO 10646-1 encoded BDF font

XKeyboardConfig-2.42

Introduction to XKeyboardConfig

The XKeyboardConfig package contains the keyboard configuration database for the X Window System.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/data/xkeyboard-config/xkeyboard-config-2.42.tar.xz>
- Download MD5 sum: 2d3b7e43e597f4c607ad6261e2b3d77f
- Download size: 892 KB
- Estimated disk space required: 20 MB
- Estimated build time: 0.1 SBU (with tests)

XKeyboardConfig Dependencies

Required

[Xorg Libraries](#)

Optional (required for tests)

[libxkbcommon-1.7.0](#), [pytest-8.3.2](#), and [Xorg Applications](#)

Installation of XKeyboardConfig

Install XKeyboardConfig by running the following commands:

```
mkdir build &&  
cd build &&  
  
meson setup --prefix=$XORG_PREFIX --buildtype=release .. &&  
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: \$XORG_PREFIX/share/X11/xkb

Xwayland-24.1.2

Introduction to Xwayland

The Xwayland package is an Xorg server running on top of the wayland server. It has been separated from the main Xorg server package. It allows running X clients inside a wayland session.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/xserver/xwayland-24.1.2.tar.xz>
- Download MD5 sum: 312c5cf17d1b147df1b787fd170958e1
- Download size: 1.2 MB
- Estimated disk space required: 28 MB (add 362 MB for tests)
- Estimated build time: 0.2 SBU (with parallelism=4; add 1.7 SBU for tests, not including clone time)

Xwayland Dependencies

Required

[libxcvt-0.1.2](#), [Pixman-0.43.4](#), [wayland-protocols-1.36](#), [Xorg Applications](#) (runtime), and [Xorg Fonts](#) (only font-util)

Recommended

[libepoxy-1.5.10](#), [libtirpc-1.3.5](#), and [Mesa-24.1.5](#)

Optional

[git-2.46.0](#) (to download packages needed for the tests), [libei-1.3.0](#), [libgcrypt-1.11.0](#), [Nettle-3.10](#), [xmlto-0.0.29](#), [Xorg Legacy Fonts](#) (only bdftopcf, for building fonts required for the tests), [rendercheck](#) (for tests), and [weston](#) (for tests)

Installation of Xwayland

Install xwayland by running the following commands:

```
sed -i '/install_man/, $d' meson.build &&

mkdir build &&
cd build &&

meson setup .. \
    --prefix=$XORG_PREFIX \
    --buildtype=release \
    -D xkb_output_dir=/var/lib/xkb &&
ninja
```

Building the test framework needs some work. First, [weston](#) brings in several dependencies, but the number can be reduced by disabling unneeded features. The `meson` command for a stripped down build of weston is shown in [Upstream continuous integration build](#).

Running the tests involves downloading two other frameworks, in addition to the mentioned optional dependencies:

```
mkdir tools &&
pushd tools &&

git clone https://gitlab.freedesktop.org/mesa/piglit.git --depth 1 &&
cat > piglit/piglit.conf << EOF
[xts]
path=$(pwd)/xts
EOF

git clone https://gitlab.freedesktop.org/xorg/test/xts --depth 1 &&

export DISPLAY=:22 &&
./hw/vfb/Xvfb $DISPLAY &
VFB_PID=$!
cd xts &&
CFLAGS=-fcommon ./autogen.sh &&
make &&
kill $VFB_PID &&
unset DISPLAY VFB_PID &&
popd
```

Then the tests can be run with:

```
XTEST_DIR=$(pwd)/tools/xts PIGLIT_DIR=$(pwd)/tools/piglit ninja test
```

Now, as the `root` user:

```
ninja install
```

If [Xorg-Server-21.1.13](#) is not installed and you do not plan to install it later, you can install `xvfb` from this package. As the `root` user:

```
install -vm755 hw/vfb/Xvfb /usr/bin
```

Command Explanations

`sed -i '/install_man/,\$d' meson.build`: Prevents installing a manual page for `xserver`, which is also provided by [Xorg-Server-21.1.13](#). Remove this command if [Xorg-Server-21.1.13](#) is not installed and you don't plan to install it later.

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: Xwayland

Installed Library: None

Installed Directory: None

Short Descriptions

`Xwayland` Allows X clients to run under wayland

Xorg-Server-21.1.13

Introduction to Xorg Server

The Xorg Server is the core of the X Window system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/xserver/xorg-server-21.1.13.tar.xz>
- Download MD5 sum: 69dbed93746d0af6b46de76d711614e4
- Download size: 4.7 MB

- Estimated disk space required: 147 MB (with tests)
- Estimated build time: 0.3 SBU (using parallelism=4; with tests)

Additional Downloads

- With the removal of the xf86-video-* drivers, the TearFree option is no longer functional. To work around this, upstream has added the TearFree option to the default modesetting driver. This patch backports this feature. Apply this patch if you are going to use Xorg in an environment without a compositor (such as TWM, IceWM, Openbox, or Fluxbox).
Optional patch: https://www.linuxfromscratch.org/patches/blfs/12.2/xorg-server-21.1.13-tearfree_backport-2.patch

Xorg Server Dependencies

Required

[libxcvt-0.1.2](#), [Pixman-0.43.4](#), [Xorg Fonts](#) (only font-util), and at runtime: [xkeyboard-config-2.42](#)

Recommended

[libepoxy-1.5.10](#) (needed for glamor), [libtirpc-1.3.5](#), [Systemd-256.4](#) (runtime), and [xorg-libinput-1.4.0](#) (runtime)

Note

Although it's possible to run the Xorg server without [Systemd-256.4](#) (rebuilt with PAM) functioning, it would require running the Xorg server as the `root` user or Xorg server will malfunction or even fail to start. The BLFS editors strongly discourage skipping this dependency. Do not attempt to do so unless you really know what you are doing.

Optional

[acpid-2.0.34](#) (runtime), [Doxygen-1.12.0](#) (to build API documentation), [fop-2.9](#) (to build documentation), [libunwind-1.6.2](#), [Nettle-3.10](#), [libgcrypt-1.11.0](#), [xcb-util-keysyms-0.4.1](#), [xcb-util-image-0.4.1](#), [xcb-util-renderutil-0.3.10](#), [xcb-util-wm-0.4.2](#) (all four to build Xephyr), [xmlto-0.0.29](#) (to build documentation), [xkeyboard-config-2.42](#) (for tests), [rendercheck](#) (for tests), and [xorg-sgml-doctools](#) (to build documentation)

Kernel Configuration

The traditional Device Dependent X (DDX) drivers have been removed from BLFS in favor of the `modesetting_drv` driver which will be built as a part of this package. To use the `modesetting_drv` driver, the kernel must provide a Direct Rendering Manager (DRM) driver for your GPU.

If your GPU supports 3D acceleration and [Mesa-24.1.5](#) provides a Gallium3D driver for utilizing its 3D capability, you should have already enabled the necessary kernel configuration options in [Mesa Kernel Configuration](#). Otherwise, you need to find the kernel configuration option of the DRM driver for the GPU and enable it. Notably, the virtual GPUs provided by some virtual machine managers:

```
Device Drivers --->
  Graphics support --->
    <*/M> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
      ...
      [DRM]
    < /*/M> DRM driver for VMware Virtual GPU           [DRM_VMWGFX]
    < /*/M> DRM Support for bochs disp1 vga interface (qemu stdvga)
      ...
      [DRM_BOCHS]
    < /*/M> Virtual Box Graphics Card                 [DRM_VBOXVIDEO]
```

If the kernel does not provide a DRM driver for your GPU, on most x86 systems the "simple frame buffer" DRM driver running on VESA or UEFI frame buffer can be used as a fallback. Enable the following options in the kernel configurations if you don't have a dedicated DRM driver for the GPU, or you want to keep the simple frame buffer driver as a fallback in case the dedicated driver fails:

```
Device Drivers --->
  Firmware Drivers --->
    [*] Mark VGA/VBE/EFI FB as generic system framebuffer      [SYSFB_SIMPLEFB]
  Graphics support --->
    <*/> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
      ...
      [DRM]
    <*/> Simple framebuffer driver                           [DRM_SIMPLEDRM]
```

To allow the kernel to print debug messages at an early boot stage, `CONFIG_DRM` and `CONFIG_DRM_SIMPLEDRM` should not be built as kernel modules unless an initramfs will be used.

If you want to use the simple frame buffer driver on a system booted via BIOS (instead of UEFI), add the following line before the first `menuentry` block in the `/boot/grub/grub.cfg` file to initialize the VESA frame buffer:

```
set gfxpayload=1024x768x32
```

You may replace `1024, 768, and 32` with a resolution and color depth setting suitable for your monitor.

If all of these DRM drivers do not work for you and you need to use a DDX driver with a non-DRM kernel GPU driver (usually named `CONFIG_FB_*` in the kernel configuration, or existing as out-tree kernel modules), or you need an device specific functionality requiring a DDX driver, consult [a prior version of BLFS](#), or [an even earlier prior version](#) for more DDX drivers.

Installation of Xorg Server

First, if you need the TearFree option to work around screen tearing, apply the backported patch:

```
patch -Np1 -i ../../xorg-server-21.1.13-tearfree_backport-2.patch
```

Install the server by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=$XORG_PREFIX \
    --localstatedir=/var \
    -D glamor=true \
    -D xkb_output_dir=/var/lib/xkb &&
ninja
```

To test the results, issue: `ninja test`. You will need to run `ldconfig` as the `root` user first or some tests may fail.

Now as the `root` user:

```
ninja install &&
mkdir -pv /etc/X11/xorg.conf.d
```

Command Explanations

`-D glamor=true`: Ensure building the Glamor module. It's needed to build the `modesetting_drv` driver which replaces the traditional Device Dependent X (DDX) drivers.

`-D suid_wrapper=true`: Builds the suid-root wrapper for legacy DDX driver support on rootless xserver systems.

`-D xephyr=true`: This option allows building Xephyr if its dependencies are met.

Contents

Installed Programs: gtf, X, Xnest, Xorg, Xvfb, and optionally Xephyr

Installed Libraries: several under `$XORG_PREFIX/lib/xorg/modules/` including the `modesetting_drv` driver

Installed Directories: `/etc/X11/xorg.conf.d`, `$XORG_PREFIX/include/xorg`, `$XORG_PREFIX/lib/xorg`, and `$XORG_PREFIX/share/X11/xorg.conf.d`

Short Descriptions

<code>gtf</code>	calculates VESA GTF mode lines
<code>x</code>	is a symbolic link to Xorg
<code>Xephyr</code>	is a nested X server which supports modern X extensions
<code>Xnest</code>	is a nested X server
<code>Xorg</code>	is the X11R7 X Server
<code>Xvfb</code>	is the virtual framebuffer X server for X Version 11
<code>modesetting_drv.so</code>	provides a video driver for machines using Kernel Mode Setting (KMS). This will use glamor if that has been enabled and the hardware offers acceleration

Xorg Input Drivers

Introduction to Xorg Input Drivers

The Xorg Input Drivers page contains the instructions for building Xorg input drivers that are necessary in order for Xorg Server to respond user inputs.

Xorg Input Drivers

- [libevdev-1.13.2](#)
- [Xorg_Evdev Driver-2.10.6](#)
- [libinput-1.26.1](#)
- [xorg-libinput-1.4.0](#)
- [Xorg_Synaptics Driver-1.9.2](#)
- [Xorg_Wacom Driver-1.2.2](#)

libevdev 1.13.2

Introduction to libevdev

The libevdev package contains common functions for Xorg input drivers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/libevdev/libevdev-1.13.2.tar.xz>
- Download MD5 sum: ddb1d798e0f2b4d0bd17c892b7d4aed3
- Download size: 452 KB
- Estimated disk space required: 6.2 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

libevdev Dependencies

Optional

[Doxygen-1.12.0](#) and [Valgrind-3.23.0](#) (optional for tests)

Kernel Configuration

Enable the following options in the kernel configuration and recompile the kernel if necessary:

```
Device Drivers --->
  Input device support --->
    -*- Generic input layer (needed for keyboard, mouse, ...)           [INPUT]
    <*/M>   Event interface                                         [INPUT_EVDEV]
```

If you want to test this package with full coverage, the following options are needed as well:

```
Device Drivers --->
  Input device support --->
    -*- Generic input layer (needed for keyboard, mouse, ...)           [INPUT]
    [*]   Miscellaneous devices --->
      <*/M>   User level driver support                                [INPUT_MISC]
                                         [INPUT_UINPUT]
```

If you build this as a module, it needs to be inserted before the test suite runs.

Installation of libevdev

Install libevdev by running the following commands:

```

mkdir build &&
cd build &&

meson setup .. \
    --prefix=$XORG_PREFIX \
    --buildtype=release \
    -D documentation=disabled &&
ninja

```

The regression tests can be run as the `root` user with `ninja test`, in a graphical session. You need to have enabled the `CONFIG_INPUT_UINPUT` setting in the kernel for full test coverage. If it is enabled as a module, the module is named `uinput` and needs to be loaded before running the tests. Note that on some systems, the tests may cause a hard lockup and require a reboot. On laptops, the system will go into Sleep and need to be woken up to finish the test suites.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Xorg Programs: libevdev-tweak-device, mouse-dpi-tool, and touchpad-edge-detector

Installed Xorg Library: libevdev.so

Installed Xorg Directory: \$XORG_PREFIX/include/libevdev-1.0

Short Descriptions

<code>libevdev-tweak-device</code>	is a tool to change some kernel device properties
<code>mouse-dpi-tool</code>	is a tool to estimate the resolution of a mouse
<code>touchpad-edge-detector</code>	touchpad-edge-detector is a tool that reads the touchpad events from the kernel and calculates the minimum and maximum for the x and y coordinates, respectively
<code>libevdev.so</code>	is a library of Xorg driver input functions

Xorg Evdev Driver-2.10.6

Introduction to Xorg Evdev Driver

The Xorg Evdev Driver package contains a Generic Linux input driver for the Xorg X server. It handles keyboard, mouse, touchpads and wacom devices, though for touchpad and wacom advanced handling, additional drivers are required.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/driver/xf86-input-evdev-2.10.6.tar.bz2>
- Download MD5 sum: e8bd1edc6751f92e425cae7eba3c61eb
- Download size: 400 KB
- Estimated disk space required: 3.9 MB
- Estimated build time: less than 0.1 SBU

Xorg Evdev Driver Dependencies

Required

[libevdev-1.13.2](#), [mtdev-1.1.7](#), and [Xorg-Server-21.1.13](#)

Installation of Xorg Evdev Driver

Install Xorg Evdev Driver by running the following commands:

```

./configure $XORG_CONFIG &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Xorg Driver:evdev_drv.so

Short Descriptions

`evdev_drv.so` is an Xorg input driver for Linux generic event devices

libinput-1.26.1

Introduction to Libinput

libinput is a library that handles input devices for display servers and other applications that need to directly deal with input devices.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/libinput/libinput/-/archive/1.26.1/libinput-1.26.1.tar.gz>
- Download MD5 sum: aae34911d662f77c1cd9be60c8502664
- Download size: 1.0 MB
- Estimated disk space required: 11 MB (add 23 MB for documentation and 9.6 MB for tests)
- Estimated build time: 0.1 SBU (add 0.1 SBU for documentation and 4.9 SBU for tests)

libinput Dependencies

Required

[libevdev-1.13.2](#) and [mtdev-1.1.7](#)

Optional

[Valgrind-3.23.0](#) (to run the tests), [GTK+-3.24.43](#) (to build the GUI event viewer), [libunwind-1.6.2](#) (required for tests), [libwacom-2.12.2](#), [sphinx-8.0.2](#) (required to build documentation), and [pyparsing-3.1.2](#) (for one non-root test)

Kernel Configuration for Running the Libinput Test Suite

Although libinput works with the same kernel configuration used by [libevdev-1.13.2](#), its extensive test suite requires the presence of `/dev/uinput` (as well as both [Valgrind-3.23.0](#) and [libunwind-1.6.2](#)).

If you wish to run the full tests, enable the following option in the kernel configuration and recompile the kernel if necessary:

```
Device Drivers --->
  Input device support --->
    -*- Generic input layer (needed for keyboard, mouse, ...)           [INPUT]
    [*]   Miscellaneous devices --->                                [INPUT_MISC]
    <*/M>   User level driver support                               [INPUT_UINPUT]
```

If you build this as a module, it needs to be inserted before the test suite runs.

On an Xorg system you will also need to prevent the events generated during the test suite from interfering with your desktop. Copy the file `test/50-litest.conf` into `$(XORG_PREFIX)/share/X11/xorg.conf.d` and restart X. For further information see [libinput test suite](#).

Installation of Libinput

Install libinput by running the following commands:

```

mkdir build &&
cd build &&

meson setup .. \
    --prefix=$XORG_PREFIX \
    --buildtype=release \
    -D debug-gui=false \
    -D tests=false \
    -D libwacom=false \
    -D udev-dir=/usr/lib/udev &&
ninja

```

Note

If you want to run the full tests, remove `-D tests=false` from the `meson` command above. Please read "kernel configuration for running the libinput test suite" (above).

If you have enabled the full tests, you can run the main tests as the `root` user by executing: `ninja test`. A very large number of tests will be run. One test fails on wayland.

Now, as the `root` user:

```
ninja install
```

If you have passed `-D documentation=true` to `meson`, you can install the generated documentation by running the following commands as the `root` user:

```

install -v -dm755      /usr/share/doc/libinput-1.26.1/html &&
cp -rv Documentation/* /usr/share/doc/libinput-1.26.1/html

```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D debug-gui=false`: This switch disables creation of a visual debug helper for libinput. Remove if you want it, and you have [GTK+-3.24.43](#) installed.

`-D tests=false`: This switch disables compilation of the main tests. Even with the tests defined as false, you can still run the first four minor tests, as a regular user, but one will be skipped if [PyParsing-3.1.2](#) is not installed.

`-D libwacom=false`: Remove this option if you have [libwacom-2.12.2](#) installed, or if you are installing GNOME.

`-D udev-dir=/usr/lib/udev`: In case that the value of `XORG_PREFIX` is not set to `/usr`, this option prevents the package from installing Udev rules and helpers into `$XORG_PREFIX/lib/udev` which is not searched by Udev daemon. This option is not needed for systems with `XORG_PREFIX` set to `/usr`, but does no harm.

`-D documentation=true`: This switch enables generation of the documentation. Add it if you want to generate the documentation. You must have [Doxygen-1.12.0](#) and [Graphviz-12.1.0](#) installed.

Contents

Installed Programs: libinput

Installed Libraries: libinput.so

Installed Directories: /etc/libinput, `$XORG_PREFIX/libexec/libinput`, `$XORG_PREFIX/share/libinput`, and (optionally) `$XORG_PREFIX/share/doc/libinput-1.26.1`

Short Descriptions

<code>libinput</code>	is a set of tools to interface with the libinput library
<code>libinput.so</code>	contains API functions for handling input devices

Xorg Libinput Driver-1.4.0

Introduction to Xorg Libinput Driver

The X.Org Libinput Driver is a thin wrapper around libinput and allows for libinput to be used for input devices in X. This driver can be used as drop-in replacement for evdev and synaptics.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/driver/xf86-input-libinput-1.4.0.tar.xz>
- Download MD5 sum: b52a1fc7456cf6595a9a1004287551a3
- Download size: 320 KB
- Estimated disk space required: 3.9 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Xorg Libinput Driver Dependencies

Required

[libinput-1.26.1](#) and [Xorg-Server-21.1.13](#)

Installation of Xorg Libinput Driver

Install Xorg Libinput Driver by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

To test the results, issue `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Xorg Driver: libinput_drv.so

Short Descriptions

`libinput_drv.so` is an Xorg input driver for mouse, keyboard, touchpad, touchscreen, and tablet devices

Xorg Synaptics Driver-1.9.2

Introduction to Xorg Synaptics Driver

The Xorg Synaptics Driver package contains the X.Org Input Driver, support programs and SDK for Synaptics touchpads. Even though the evdev driver can handle touchpads very well, this driver is required if you want to use advanced features like multi tapping, scrolling with touchpad, turning the touchpad off while typing, etc.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/driver/xf86-input-synaptics-1.9.2.tar.xz>
- Download MD5 sum: 3b95e7baf4428b114e9910f999e96601
- Download size: 424 KB
- Estimated disk space required: 5.2 MB
- Estimated build time: less than 0.1 SBU

Xorg Synaptics Driver Dependencies

Required

[libevdev-1.13.2](#) and [Xorg-Server-21.1.13](#)

Installation of Xorg Synaptics Driver

Install Xorg Synaptics Driver by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: synclient and syndaemon

Installed Xorg Driver: synaptics_drv.so

Short Descriptions

<code>synclient</code>	is a command line utility used to query and modify Synaptics driver options
<code>syndaemon</code>	is a program that monitors keyboard activity and disables the touchpad when the keyboard is being used
<code>synaptics_drv.so</code>	is an Xorg input driver for touchpads

Xorg Wacom Driver-1.2.2

Introduction to Xorg Wacom Driver

The Xorg Wacom Driver package contains the X.Org X11 driver and SDK for Wacom and Wacom-like tablets. It is not required to use a Wacom tablet, the xf86-input-evdev driver can handle these devices without problems.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/linuxwacom/xf86-input-wacom/releases/download/xf86-input-wacom-1.2.2/xf86-input-wacom-1.2.2.tar.bz2>
- Download MD5 sum: bd72f813b094b79b450e50a95476977b
- Download size: 636 KB
- Estimated disk space required: 8.1 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Xorg Wacom Drivers Dependencies

Required

[Xorg-Server-21.1.13](#)

Optional

[Doxygen-1.12.0](#) and [Graphviz-12.1.0](#)

Kernel Configuration

To use a Wacom tablet with USB interface, enable the following options in your kernel configuration and recompile. Note that other configuration options could be required for tablet with a serial or bluetooth interface:

```
Device Drivers --->  
[*] HID bus support ---> [HID_SUPPORT]
```

```

{*/M} HID bus core support [HID]
  Special HID drivers --->
<*/M>   Wacom Intuos/Graphire tablet support (USB) [HID_WACOM]
  USB HID support --->
<*/M>   USB HID transport layer [USB_HID]
[*] USB support ---> [USB_SUPPORT]
  <*/M> Support for Host-side USB [USB]

```

Installation of Xorg Wacom Driver

Install Xorg Wacom Driver by running the following commands:

```

./configure $XORG_CONFIG &&
make

```

To test the results, issue: `make check`.

Now, as the `root` user:

```

make install

```

Contents

Installed Programs: isdv4-serial-debugger, isdv4-serial-inputattach, and xsetwacom

Installed Xorg Driver: wacom_drv.so

Short Descriptions

<code>xsetwacom</code>	is a commandline utility used to query and modify wacom driver settings
<code>wacom_drv.so</code>	is an Xorg input driver for Wacom devices

twm-1.0.12

Introduction to twm

The twm package contains a very minimal window manager.

This package is provided for testing the completed Xorg installation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/app/twm-1.0.12.tar.xz>
- Download MD5 sum: 805ee08b5a87e1103dfe2eb925b613b4
- Download size: 260 KB
- Estimated disk space required: 3.8 MB
- Estimated build time: less than 0.1 SBU

twm Dependencies

Required

[Xorg-Server-21.1.13](#)

Installation of twm

Install twm by running the following commands:

```

sed -i -e '/^rcdir = /s,\^(\rcdir = \).*,\1/etc/X11/app-defaults,' src/Makefile.in &&
./configure $XORG_CONFIG &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed -i -e '/^rcdir =/s...':` This command ensures the twm configuration file gets installed in the proper location.

Contents

Installed Programs: twm

Installed Libraries: None

Installed Directory: /etc/X11/app-defaults

Short Descriptions

`twm` is the Tab Window Manager for the X Window System

xterm-393

Introduction to xterm

xterm is a terminal emulator for the X Window System.

This package is provided for testing the completed Xorg installation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://invisible-mirror.net/archives/xterm/xterm-393.tgz>
- Download MD5 sum: 1ac5147362addade51c4e627d8cb328f
- Download size: 1.5 MB
- Estimated disk space required: 15 MB
- Estimated build time: 0.1 SBU (with parallelism=4)

xterm Dependencies

Required

[luit-20240102](#)

Required (at runtime)

A monospace TTF or OTF font such as [Dejavu fonts](#)

Optional

[Emacs-29.4](#), [pcre2-10.44](#), [Valgrind-3.23.0](#) and [man2html](#)

Installation of xterm

Install xterm by running the following commands:

```
sed -i '/v0/{n;s/new:/new:kb=^?:/}' termcap &&
printf '\tkbs=\\\177,\n' >> terminfo &&

TERMINFO=/usr/share/terminfo \
./configure $XORG_CONFIG \
--with-app-defaults=/etc/X11/app-defaults &&

make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
mkdir -pv /usr/share/applications &&
cp -v *.desktop /usr/share/applications/
```

Command Explanations

`sed -i ... termcap, printf ... >> terminfo`: These commands modify the terminal description so that the Backspace key is expected to send the character with ASCII code 127. This is done for consistency with the Linux console.

`TERMINFO=/usr/share/terminfo`: This ensures that the `xterm` terminfo files are installed to the system terminfo database.

`--with-app-defaults=...`: Sets the location for the `app-defaults` directory.

Configuring xterm

There are two ways to configure xterm. You can add the X resource definitions to the user's `~/.Xresources` file, or add them to the system-wide `$XORG_PREFIX/share/X11/app-defaults/Xterm` file.

In order for xterm to follow the locale settings in the environment, use TrueType fonts, and follow the Linux convention about the code sent by the Backspace key, add the following definitions as the `root` user:

```
cat >> /etc/X11/app-defaults/XTerm << "EOF"
*VT100*locale: true
*VT100*faceName: Monospace
*VT100*faceSize: 10
*backarrowKeyIsErase: true
*ptyInitialErase: true
EOF
```

Contents

Installed Programs: koi8rxterm, resize, uxterm, and xterm

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>koi8rxterm</code>	is a wrapper script to set up xterm with a KOI8-R locale
<code>resize</code>	prints a shell command for setting the TERM and TERMCAP environment variables to indicate the current size of xterm window
<code>uxterm</code>	is a wrapper script that modifies the current locale to use UTF-8 and starts xterm with the proper settings
<code>xterm</code>	is a terminal emulator for the X Window System

xclock-1.1.1

Introduction to xclock

The xclock package contains a simple clock application which is used in the default xinit configuration.

This package is provided for testing the completed Xorg installation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/app/xclock-1.1.1.tar.xz>
- Download MD5 sum: 1273e3f4c85f1801be11a5247c382d07
- Download size: 156 KB
- Estimated disk space required: 1.5 MB
- Estimated build time: less than 0.1 SBU

xclock Dependencies

Required

[Xorg Libraries](#)

Installation of xclock

Install xclock by running the following commands:

```
./configure $XORG_CONFIG &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: xclock

Installed Libraries: None

Installed Directory: None

Short Descriptions

`xclock` is an analog/digital clock for X

xinit-1.4.2

Introduction to xinit

The xinit package contains a usable script to start the xserver.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/app/xinit-1.4.2.tar.xz>
- Download MD5 sum: 0e9a1b9a82b84ab229c709c0f939c113
- Download size: 153 KB
- Estimated disk space required: 1.5 MB
- Estimated build time: less than 0.1 SBU

xinit Dependencies

Required

[Xorg Libraries](#)

Recommended (runtime only)

[twm-1.0.12](#), [xclock-1.1.1](#), and [xterm-393](#) (used in the default `xinitrc` file)

Installation of xinit

Install xinit by running the following commands:

```
./configure $XORG_CONFIG --with-xinitedir=/etc/X11/app-defaults &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
ldconfig
```

Contents

Installed Programs: xinit and startx

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>startx</code>	initializes an X session
<code>xinit</code>	is the X Window System initializer

Xorg-7 Testing and Configuration

Testing Xorg

Note

Before starting Xorg for the first time, it is useful to rebuild the library cache by running `ldconfig` as the `root` user.

Note

Before starting Xorg for the first time, it is often needed to reboot the system to ensure all appropriate daemons are started and appropriate security issues are properly set. As an alternative, logging out and logging back in may work, but as of this writing has not been tested.

Warning

If Xorg hangs for some reason (for example, lacking a proper input driver), the system may stop responding to any user input. As a precaution, you can enable a magic **SysRq** key before testing Xorg. As the `root` user, issue:

```
echo 4 > /proc/sys/kernel/sysrq
```

Then if Xorg hangs, it's possible to use **Alt+SysRq+R** to reset the keyboard mode. Now it should be able to use **Ctrl+Alt+Fx** (replace x with a VT number) to switch to another VT. If it works, login and kill Xorg using command line in the new VT.

To test the Xorg installation, issue `startx`. This command brings up a rudimentary window manager called `twm` with three xterm windows and one xclock window. The xterm window in the upper left is a login terminal and running `exit` from this terminal will exit the X Window session. The third xterm window may be obscured on your system by the other two xterms.

Note

When testing Xorg with the `twm` window manager, there will be several warnings in the Xorg log file, `$HOME/.local/share/xorg/Xorg.0.log`, about missing font files. In addition, there will be several warnings on the text mode terminal (usually `tty1`) about missing fonts. These warnings do not affect functionality, but can be removed if desired by installing the [Xorg Legacy Fonts](#).

Note

On systems with NVIDIA GPUs which are using the Nouveau kernel driver, you may encounter occasional GPU crashes and hangs. If this problem occurs, downgrade to the latest version of the Linux 6.1 kernel.

Generally, there is no specific configuration required for Xorg, but customization is possible. For details, see [the section called "Setting up Xorg Devices"](#) below.

Checking the Direct Rendering Infrastructure (DRI) Installation

DRI is a framework for allowing software to access graphics hardware in a safe and efficient manner. It is installed in X by default (using Mesa) if you have a supported video card.

To check if DRI drivers are installed properly, check the log file `$HOME/.local/share/xorg/Xorg.0.log` (or `/var/log/Xorg.0.log` if you have built [Xorg-Server-21.1.13](#) with the uid bit) for statements such as:

```
(II) modeset(0): [DRI2] Setup complete
(II) modeset(0): [DRI2]     DRI driver: crocus
(II) modeset(0): [DRI2]     VDPAU driver: va_gl
```

Note

DRI configuration may differ if you are using alternate drivers, such as traditional DDX drivers, or the proprietary drivers from [NVIDIA](#) or [AMD](#).

Another way to determine if DRI is working properly is to use one of the two optionally installed OpenGL demo programs in [Mesa-24.1.5](#). From an X terminal, run `glxinfo -B` and look for the phrase:

```
name of display: :0
display: :0 screen: 0
direct rendering: Yes
```

If direct rendering is enabled, you can add verbosity by running `LIBGL_DEBUG=verbose glxinfo`. This will show the drivers, device nodes and files used by the DRI system.

To confirm that DRI2 hardware acceleration is working, you can (still in the X terminal) run the command `glxinfo | grep -E "(OpenGL vendor|OpenGL renderer|OpenGL version)"`. If that reports something *other than* Software Rasterizer then you have working acceleration for the user who ran the command.

If your hardware does not have any DRI2 driver available, it will use a Software Rasterizer for Direct Rendering. In such cases, you can use a new, LLVM-accelerated, Software Rasterizer called LLVMPipe. In order to build LLVMPipe just make sure that [LLVM-18.1.7](#) is present at Mesa build time. Note that all decoding is done on the CPU instead of the GPU, so the display will run slower than with hardware acceleration. To check if you are using LLVMPipe, review the output of the `glxinfo` command above. An example of the output using the Software Rasterizer is shown below:

```
OpenGL vendor string: VMware, Inc.
OpenGL renderer string: Gallium 0.4 on llvmpipe (LLVM 3.5, 256 bits)
OpenGL version string: 3.0 Mesa 10.4.5
```

You can also force LLVMPipe by exporting the `LIBGL_ALWAYS_SOFTWARE=1` environment variable when starting Xorg.

Again, if you have built the Mesa OpenGL demos, you can also run the test program `glxgears`. This program brings up a window with three gears turning. The X terminal will display how many frames were drawn every five seconds, so this will give a rough benchmark. The window is scalable, and the frames drawn per second is highly dependent on the size of the window. On some hardware, `glxgears` will run synchronized with the vertical refresh signal and the frame rate will be approximately the same as the monitor refresh rate.

Debugging Xorg

When starting xorg, there are a couple of ways to check what any issues you may have. If the system comes up, you can see what driver is being used by running `xdriinfo`. If there are issues or you just want to check, look at `Xorg.0.log`.

The location of `Xorg.0.log` depends on how Xorg is installed. If the instructions in the book are followed closely and Xorg is started from the command line, it will be located in the `$HOME/.local/share/xorg/` directory. If Xorg is started by a display manager (e.g. [lightdm-1.32.0](#), [ssdm-0.21.0](#), or [GDM-46.2](#)) or if `$XORG_PREFIX/bin/Xorg` has the uid bit set, it will be located in the `/var/log/` directory.

Xorg.0.log Issues

When you look at `Xorg.0.log`, check for entries like (EE) or (WW). Below are some common entries:

(WW) Open ACPI failed (/var/run/acpid.socket)

This warning is because [acpid-2.0.34](#) is not installed. If you are not on a laptop, it can be safely ignored. On a laptop, install [acpid-2.0.34](#) to enable actions like recognizing when the lid is closed.

(WW) VGA arbiter: cannot open kernel arbiter, no multi-card support

This warning is displayed when a regular user starts Xorg. The library `libpciaccess.so` issues this warning when it tries to open `/dev/vga_arbiter`. If there is no more than one legacy PCI (not PCIe) graphic cards on the system, it can safely be ignored. If really necessary, the permissions of this device can be changed by adding a udev rule and adding the local user to the video group. As the `root` user:

```
cat > /etc/udev/rules.d/99-vga-arbiter.rules << EOF
# /etc/udev/rules.d/99-vga-arbiter.rules: Set vga_arbiter group/mode

ACTION=="add", KERNEL=="vga_arbiter", GROUP="video" MODE="0660"
EOF

usermod -a -G video <user running xorg>
```

Hybrid Graphics

Hybrid Graphics is still in experimental state for Linux. Xorg Developers have developed a technology called PRIME that can be used for switching between integrated and muxless discrete GPU at will. Automatic switching is not possible at the moment.

In order to use PRIME for GPU switching, make sure that you are using Linux Kernel 3.4 or later (recommended). You will need latest DRI and DDX drivers for your hardware and Xorg Server 1.13 or later.

Xorg Server should load both GPU drivers automatically. You can check that by running:

```
xrandr --listproviders
```

There should be two (or more) providers listed, for example:

```
Providers: number : 2
Provider 0: id: 0x7d cap: 0xb, Source Output, Sink Output, Sink Offload crtcs: 3 outputs: 4 associated providers: 1 name:modesetting
Provider 1: id: 0x56 cap: 0xf, Source Output, Sink Output, Source Offload, Sink Offload crtcs: 6 outputs: 1 associated providers: 1 n
```

In order to be able to run a GLX application on a discrete GPU, you will need to run the following command, where `<provider>` is the id of the more powerful discrete card, and `<sink>` is the id of card which has a display connected:

```
xrandr --setprovideroffloadsink <provider> <sink>
```

Note

With the Xorg modesetting driver, which is DRI3 capable, the above command is no longer necessary. It does no harm however.

Then, you will need to export the `DRI_PRIME=1` environment variable each time you want the powerful GPU to be used. For example,

```
DRI_PRIME=1 glxinfo | grep -E "(OpenGL vendor|OpenGL renderer|OpenGL version)"
```

will show OpenGL vendor, renderer and version for the discrete GPU.

If the last command reports same OpenGL renderer with and without `DRI_PRIME=1`, you will need to check your installation.

Setting up Xorg Devices

For most hardware configurations, modern Xorg will automatically get the server configuration correct without any user intervention. There are, however, some cases where auto-configuration will be incorrect. Following are some example manual configuration items that may be of use in these instances.

Setting up X Input Devices

For most input devices, no additional configuration will be necessary. This section is provided for informational purposes only.

A sample default XKB setup could look like the following (executed as the `root` user):

```
cat > /etc/X11/xorg.conf.d/xkb-defaults.conf << "EOF"
Section "InputClass"
    Identifier "XKB Defaults"
    MatchIsKeyboard "yes"
    Option "XkbLayout" "fr"
    Option "XkbOptions" "terminate:ctrl_alt_bksp"
EndSection
EOF
```

The "XkbLayout" line is an example for a French (AZERTY) keyboard. Change it to your keyboard model. That line is not needed for a QWERTY (US) keyboard.

Fine Tuning Display Settings

If you want to set the monitor resolution for Xorg, first run `xrandr` in a X terminal to list the supported resolutions and the corresponding refresh rates. For example, it outputs the following for one monitor:

```
Screen 0: minimum 16 x 16, current 5760 x 2160, maximum 32767 x 32767
DP-1 connected primary 3840x2160+0+0 (normal left inverted right x axis y axis) 600mm x 340mm
  3840x2160      59.98+*
  2048x1536      59.95
  1920x1440      59.90
  1600x1200      59.87
  1440x1080      59.99
  1400x1050      59.98
  1280x1024      59.89
  1280x960       59.94
  1152x864       59.96
  1024x768       59.92
  800x600        59.86
  640x480        59.38
```

From the output we can see the monitor is identified `DP-1`. Select a suitable resolution from the output list, for example `1920x1440`. Then as the `root` user, create a configuration file for the Xorg server:

```
cat > /etc/X11/xorg.conf.d/monitor-DP-1.conf << "EOF"
Section "Monitor"
    Identifier "DP-1"
    Option     "PreferredMode" "1920x1440"
EndSection
EOF
```

Sometimes `xrandr` may fail to detect some resolution settings supported by the monitor. It usually happens with virtual monitors or virtual machine managers like [qemu-9.0.2](#) or VMWare: a virtual monitor actually supports all pairs of integers in a range as the resolution, but `xrandr` will only list a few. To use a resolution not listed by `xrandr`, first run `cvt` to get the mode line for the resolution. For example:

```
cvt 1600 900
# 1600x900 59.95 Hz (CVT 1.44M9) hsync: 55.99 kHz; pclk: 118.25 MHz
Modeline "1600x900_60.00" 118.25 1600 1696 1856 2112 900 903 908 934 -hsync +vsync
```

As the `root` user, create a Xorg server configuration file containing this mode line, and specify the mode as preferred mode:

```
cat > /etc/X11/xorg.conf.d/monitor-DP-1.conf << "EOF"
Section "Monitor"
    Identifier "DP-1"
    Modeline   "1600x900_60.00" 118.25 1600 1696 1856 2112 900 903 908 934 -hsync +vsync
    Option     "PreferredMode" "1600x900_60.00"
EndSection
EOF
```

Some high-end LCD monitors support a refresh rate higher than 100 Hz but `xrandr` may fail to recognize the supported refresh rate and use 60 Hz instead. This issue would prevent you from utilizing the full capability of the monitor, and may cause the screen to flicker or show "artifacts" like meshes or grids. To resolve the issue, again use `cvt` to get the mode line with a custom refresh rate:

```
cvt 3840 2160 144
# 3840x2160 143.94 Hz (CVT) hsync: 338.25 kHz; pclk: 1829.25 MHz
Modeline "3840x2160_144.00" 1829.25 3840 4200 4624 5408 2160 2163 2168 2350 -hsync +vsync
```

Then paste it into the Xorg server configuration file and set it as the preferred mode.

Another common setup is having multiple server layouts for use in different environments. Though the server will automatically detect the presence of another monitor, it may get the order incorrect:

```
cat > /etc/X11/xorg.conf.d/server-layout.conf << "EOF"
Section "ServerLayout"
    Identifier      "DefaultLayout"
    Screen         0  "Screen0" 0 0
    Screen         1  "Screen1" LeftOf "Screen0"
    Option          "Xinerama"
EndSection
EOF
```

When you drag a window in twm (or any non-compositing window manager) horizontally, you may observe that the vertical borders of the window are broken into multiple segments. This is an example of the visual artifacts called *screen tearing*. To resolve the screen tearing problems, create a configuration file that enables the TearFree option. Note that you must have the Tearfree patch applied from [Xorg-Server-21.1.13](#) for this to function properly, and it may increase memory allocation and reduce performance.

```
cat > /etc/X11/xorg.conf.d/20-tearfree.conf << "EOF"
Section "Device"
    Identifier "Graphics Adapter"
    Driver     "modesetting"
    Option     "TearFree" "true"
EndSection
EOF
```

With modern Xorg, little or no additional graphic card configuration is necessary. If you should need extra options passed to your video driver, add them into the `Device` section as well. The options supported by the modesetting driver are documented in the man page [modesetting\(4\)](#).

Tuning Fontconfig

Overview of Fontconfig

If you only read text in English, and are happy with the common libre fonts listed on the next page, you may never need to worry about the details of how Fontconfig works. But there are many things which can be altered if they do not suit your needs.

However, before spending a long time trying out different settings please be aware that modern applications and desktop environments may ignore the settings of fontconfig - see [Items which can override Fontconfig](#).

Although this page is long, it barely scratches the surface and you will be able to find many alternative views on the web (but please remember that some things have changed over the years, for example the autohinter is no longer the default). The aim here is to give you enough information to understand the changes you are making, why they may not always work, and to identify online information which is no-longer appropriate.

Unfortunately, some of the terminology is ambiguous (e.g. 'font face' can mean a name known to Fontconfig, or the ordinary, condensed, etc variations of a font) and 'style' can be used to differentiate 'ordinary' from 'italic', or in describing some classes of Serif fonts.

The following links are to assist navigation in this page.

- [The Xft Font Protocol](#)
- [Useful Commands](#)
- [The configuration files](#)
- [The rules to choose a font](#)
- [Hinting and Anti-aliasing](#)
- [Disabling Bitmap fonts](#)
- [Synthetic changes](#)
- [Adding extra font directories](#)
- [Preferring certain fonts](#)
- [Fontconfig user documentation](#)
- [Prefer a specific font](#)

- [Preferring chosen CJK fonts](#)
- [Editing Old-Style conf files](#)
- [About font weights](#)
- [Items which can override Fontconfig](#)
- [External Links](#)

The Xft Font Protocol

The Xft font protocol provides antialiased font rendering through freetype, and fonts are controlled from the client side using Fontconfig (except for [rxvt-unicode-9.31](#) which can use fonts listed in `~/.Xresources`, and [AbiWord-3.0.5](#) which only uses the specified font). The default search path is `/usr/share/fonts` and `~/.local/share/fonts`, although for the moment the old and deprecated location `~/.fonts` still works. Fontconfig searches directories in its path recursively and maintains a cache of the font characteristics in each directory. If the cache appears to be out of date, it is ignored, and information is fetched from the fonts themselves (that can take a few seconds if you have a lot of fonts installed).

If you've installed Xorg in any prefix other than `/usr`, any X fonts were not installed in a location known to Fontconfig. Symlinks were created from the `OTF` and `TTF` X font directories to `/usr/share/fonts/X11-{OTF,TTF}` in Xorg Fonts. This allows Fontconfig to use the OpenType and TrueType fonts provided by X, although many people will prefer to use more modern fonts.

Fontconfig uses names to define fonts. Applications generally use generic font names such as "Monospace", "Sans" and "Serif". Fontconfig resolves these names to a font that has all characters that cover the orthography of the language indicated by the locale settings.

Useful Commands

The following commands may be helpful when working with Fontconfig, particularly if you are interested in overriding which font will be chosen.

`fc-list | less` : shows a list of all available fonts (`/path/to/filename: Font Name:style`). If you installed a font and it doesn't show, then the directory it is contained in is not readable by your user.

`fc-match 'Font Name'` : tells you which font will be used if the named font is requested. Typically you would use this to see what happens if a font you have not installed is requested, but you can also use it if the system is giving you a different font from what you expected (perhaps because Fontconfig does not think that the font supports your language).

`fc-match type` : shows which font will be used in the current language for that `TYPE` (Monospace, Sans, Serif (*initial capital letter is optional*)). If that font does not map a codepoint, Fontconfig can take a glyph from any available font, even if it is not of the specified type. Any other value for `TYPE` will be assumed to be Sans.

`fc-match 'Serif :lang=ja:weight=bold'` will tell you which font and weight will be chosen for Japanese text in bold weight. It does not mean that the reported font will necessarily be able to show Japanese ideograms, so a fallback might be used, or some glyphs may be missing. For language codes, use ISO-639 values such as 'fr', 'ja', 'zh-cn'. Note that an unrecognized value such as just 'zh' will not return any match. To illustrate the fallback, on a system where both Noto Sans Mono and DejaVu Sans Mono are installed, `fc-match 'monospace :lang=en` shows Noto Sans Mono will be used, but if the lang is changed to 'ar' (arabic) DejaVu Sans will be used.

If you want to determine if a font file has hinting (many older fonts do not, because it was patented) use `fc-query /path/to/fontfile | grep 'fonthashint:'` which will report 'True(s)' or 'False(s)'. Some recent fonts with both TTF and OTF versions will have hinting in the TTF files.

If you wish to know which font will be used for a string of text (i.e. one or more glyphs, preceded by a space), paste the following command and replace the `xyz` by the text you care about:

`FC_DEBUG=4 pango-view --font=monospace -t xyz | grep family` : this requires [Pango-1.54.0](#) and [ImageMagick-7.1.1-36](#) - it will invoke `display` to show the text in a tiny window, and after closing that the last line of the output will show which font was chosen. This is particularly useful for CJK languages, and you can also pass a language, e.g. `PANGO_LANGUAGE=en;ja` (English, then assume Japanese) or just `zh-cn` (or other variants such as `zh-sg` or `zh-tw`).

The configuration files

The main files are in `/etc/fonts/conf.d/`, which was intended to be a directory populated by symlinks to some of the files in `/usr/share/fontconfig/conf.avail/`. But many people, and some packages, create the files directly. Each file name must be in the form of two digits, a dash, somename.conf and they are read in sequence.

By convention, the numbers are assigned as follows:

- 00-09 extra font directories
- 10-19 system rendering defaults (such as antialiasing)

- 20-29 font rendering options
- 30-39 family substitution
- 40-49 map family to generic type
- 50-59 load alternate config files
- 60-69 generic aliases, map generic to family
- 70-79 adjust which fonts are available
- 80-89 match target scan (modify scanned patterns)
- 90-99 font synthesis

You can also have a personal `fonts.conf` in `$XDG_CONFIG_HOME` (which is `~/.config/fontconfig/`).

The rules to choose a font

If the requested font is installed, and provided it contains the codepoints *required* for the current language (in the source, see the `.orth` files in the `fc-lang/` directory), it will be used.

However, if the document or page requested a font which is not installed (or, occasionally, does not contain all the required codepoints) the following rules come into play: First, `30-metric-aliases.conf` is used to map aliases for some fonts with the same metrics (same size, etc). Note that there are both weak and strong aliases so that aliases for one form such as Helvetica or Times New Roman can be satisfied by the other style, i.e. anything which is an alias of Arial or Times in those examples. Some examples of Latin fonts with the same metrics can be found in the 'Substitutes' PDFs at typosetting.co.uk.

After that, an unknown font will be searched for in `45-latin.conf`: 'Latin' covers Cyrillic and Greek, and now also maps system-ui fonts which are used for User Interface messages in other alphabets. If the font is found it will be mapped as serif, sans-serif, monospace, fantasy, cursive, or system-ui. Otherwise, `49-sansserif.conf` will assume it is Sans.

Then `60-latin.conf` provides ordered lists of the fallbacks - [Noto fonts](#) will be used if you installed them. Cyrillic and Greek appear to be treated in the same way. All of these files prefer commercial fonts if they are present, although modern libre fonts are often at least equal. Finally, if a codepoint is still not found it can be taken from any available system font. The following details only mention freely available fonts.

Default Persian fonts are dealt with in `65-fonts-persian.conf`. It looks as if all the listed fonts are commercial. Using fonts that support Persian (which has its own variant of the arabic alphabet, and its own font styles) is outside the skills of the BLFS editors.

All remaining scripts for which Fontconfig has preferences (CJK scripts, Indic scripts) are dealt with in `65-nonlatin.conf`. These are again nominally grouped as Serif, Sans-Serif, Monospace. Of the free fonts, WenQuanYi Zen Hei (Pan-CJK Sans) comes first for both Serif and for Sans. Therefore, if you install this as a fallback but want to use different fonts for Japanese or Korean you will need to set up a preference. Similarly, the old fireflysung Serif font is also listed for Sans.

After Pan-CJK and Chinese fonts come several Japanese fonts and then several Korean fonts (both split appropriately between Sans and Serif). Finally come the various Lohit Indic families (one font file per script), labelled as both Sans and Serif.

The Monospace fonts listed in `65-nonlatin.conf` do not include WenQuanYi Zen Hei although that will be available as a fallback if installed. Several Japanese Gothic fonts are listed, followed by AR PL KaitiM GB (a zh-sc 'Brush' font), AR PL Serif fonts for zh-sc (SungtiL) and zh-tw (Mingti2L), some Korean Sans fonts and the various Lohit Indic families.

For UI fonts, various Noto Sans UI fonts are the only listed free fonts.

The various Noto CJK fonts are *not* among the listed fonts, possibly the RedHat developers preferred other fonts. These now come in many variations, and most users who use these will not install any other CJK fonts.

Before Fontconfig-2.14, the first preferred Latin font family was Bitstream Vera. In practice that was rarely used because it covered so little. After that, DejaVu was the next preferred family, so people were recommended to install that. That has now changed, Bitstream Vera has been replaced by the relevant Noto fonts (Serif, Sans, Sans Mono), so these will be preferred if they have been installed, followed by DejaVu.

For serif, Times New Roman could have been aliased from Liberation Serif or Tinos, and Times from TeX Gyre Termes, so although the named fonts are not free, the metric-compatible fonts can be used. Ignoring other non-free fonts, the remaining order for serif is: Times New Roman, Luxi Serif, Nimbus Roman No9 L, and Times. In practice, that means those fonts at the end of the list are unlikely to be used unless a web page asks for them.

For sans-serif, the remaining order is anything mapped to Arial, Luxi Sans, Nimbus Sans L, and anything mapped to Helvetica.

The remaining alternatives for monospace are Inconsolata, anything mapped to Courier New, Luxi Mono, Nimbus Mono, and anything mapped to Courier.

For 'fantasy' there are no free fonts, so Fontconfig will fall back to sans-serif.

For 'cursive', the only free font is TeX Gyre Chorus as an alias for ITC Zapf chancery, otherwise Fontconfig will again fall back to sans-serif.

The system-ui category is unusual. It is for interface messages, so some scripts need special versions to fit in the available space. For Latin, Greek and Cyrillic an ordinary sans font should fit without problems. However, the first preferred font is Cantarell, followed by Noto Sans UI. Cantarell started as a Latin sans-serif font, that has been forked in Gnome under the same name but they only provide the source. The Noto Sans UI fonts are for other languages.

Since Fontconfig-2.12.5, there is also generic family matching for some emoji and math fonts, please see {45,60}-generic.conf.

In the rare cases where a font does not contain all the expected codepoints, see 'Trial the First:' at [I stared into the fontconfig](#) for the long details.

Hinting and Anti-aliasing

It is possible to change how, or if, fonts are hinted. The following example file contains the default settings, but with comments. The settings are very much down to the user's preferences and to the choice of fonts, so a change which improves some pages may worsen others. The preferred location for this file is: `~/.config/fontconfig/fonts.conf`

To try out different settings, you may need to exit from Xorg and then run `startx` again so that all applications use the new settings. Several things can override the fontconfig settings, see [Items which can override Fontconfig](#) below for more details. To explore the possibilities, create a file for your user:

```
mkdir -pv ~/.config/fontconfig &&
cat > ~/.config/fontconfig/fonts.conf << "EOF"
<?xml version='1.0'?>
<!DOCTYPE fontconfig SYSTEM 'fonts.dtd'>
<fontconfig>

<match target="font" >
    <!-- autohint was the old automatic hinter when hinting was patented
        so turn it off to ensure any hinting information in the font
        itself is used, this is the default -->
    <edit mode="assign" name="autohint"> <bool>false</bool></edit>

    <!-- hinting is enabled by default -->
    <edit mode="assign" name="hinting"> <bool>true</bool></edit>

    <!-- for the lcdfilter see https://www.spasche.net/files/lcdfiltering/ -->
    <edit mode="assign" name="lcdfilter"> <const>lcddefault</const></edit>

    <!-- options for hintstyle:
        hintfull: is supposed to give a crisp font that aligns well to the
        character-cell grid but at the cost of its proper shape. However, anything
        using Pango >= 1.44 will not support full hinting, Pango now uses harfbuzz
        for hinting. Apps which use Skia (e.g. Chromium, Firefox) should not be
        affected by this.

        hintmedium: is reported to be broken.
        hintslight is the default: - supposed to be more fuzzy but retains shape.

        hintnone: seems to turn hinting off.
        The variations are marginal and results vary with different fonts -->
    <edit mode="assign" name="hintstyle"> <const>hintslight</const></edit>

    <!-- antialiasing is on by default and really helps for faint characters
        and also for 'xft:' fonts used in rxvt-unicode -->
    <edit mode="assign" name="antialias"> <bool>true</bool></edit>

    <!-- subpixels are usually rgb, see
        http://www.lagom.nl/lcd-test/subpixel.php -->
    <edit mode="assign" name="rgba"> <const>rgb</const></edit>

    <!-- thanks to the Arch wiki for the lcd and subpixel links -->
</match>

</fontconfig>
EOF
```

You will now need to edit the file in your preferred editor. Many of the different settings give very subtle differences and the results may differ for some of the fonts you use.

Note

Hinting, if enabled, is done in FreeType. Since FreeType-2.7 the default TrueType interpreter is v40. The original v35 hinter could be enabled by an environment variable, but is only really appropriate to original Microsoft TTF fonts (Arial, etc). The v38 hinter (Infinality) is not built by default and all the options to tune it have been removed. For full details see [subpixel-hinting](#). Spoiler: there is NO sub-pixel hinting, the code simply ignores *all* horizontal hinting instructions.

Xorg assumes screens have 96 dots per inch (DPI). Most LCD screens are close to this, but some people detect colour fringing if their screen diverges from that size. See [calc-dpi](#).

If you have a High DPI screen (often described as '4K' or larger) you will probably use larger font sizes and benefit from disabling hinting.

For more examples see the blfs-support thread which started at [2016-09/00128](#), particularly [2016-09/00137](#), and the original poster's preferred solution at [2016-09/00147](#). There are other examples in [Fontconfig in the Arch wiki](#) and [Fontconfig in the Gentoo wiki](#).

Disabling Bitmap Fonts

In previous versions of BLFS, the ugly old Xorg bitmap fonts were installed. Now, many people will not need to install any of them. But if for some reason you have installed one or more bitmap fonts, you can prevent them from being used by Fontconfig by creating the following file as the `root` user :

```
cat > /etc/fonts/conf.d/70-no-bitmaps.conf << "EOF"
<?xml version='1.0'?>
<!DOCTYPE fontconfig SYSTEM 'fonts.dtd'>
<fontconfig>
<!-- Reject bitmap fonts -->
<selectfont>
<rejectfont>
<pattern>
  <patelt name="scalable"><bool>false</bool></patelt>
</pattern>
</rejectfont>
</selectfont>
</fontconfig>
EOF
```

Synthetic changes

In `90-synthetic.conf` there are examples of applying synthetic slanting and emboldening to a font. The synthetic emboldening can be applied to a visibly faint font, but the results are not always as expected: With just the embolden, Epiphany showed darker fonts while Firefox did not - so although Cairo is now used by firefox the comment about setting Weight is still valid. But setting both, Epiphany will show bold text by default, but it will show heavy text if markup for bold is used. In both cases, neither libreOffice nor falkon showed bolder text.

Adding extra font directories

Normally, system fonts and user fonts are installed in directories beneath the locations specified in [The Xft Font Protocol](#) and there is no obvious reason to put them elsewhere. However, a full BLFS install of [texlive-20240312](#) puts many fonts in `/opt/texlive/2024/texmf-dist/fonts/` in the `opentype/` and `truetype/` subdirectories. Although pulling in all of these files may appear useful (it allows you to use them in non TeX programs), there are several problems with such an approach:

1. There are hundreds of files, which makes selecting fonts difficult.
2. Some of the files do odd things, such as displaying semaphore flags instead of ASCII letters, or mapping cyrillic codepoints to character forms appropriate to Old Church Slavonic instead of the expected current shapes: fine if that is what you need, but painful for normal use.
3. Several fonts have multiple sizes and impenetrable short names, which both make selecting the correct font even more difficult.
4. When a font is added to CTAN, it is accompanied by TeX packages to use it in the old engines (xelatex does not normally need this), and then the version is often frozen whilst the font is separately maintained. Some of these fonts such as [Dejavu fonts](#) are probably already installed on your BLFS system in a newer version, and if you have multiple versions of a font it is unclear which one will be used by Fontconfig.

However, it is sometimes useful to look at these fonts in non-TeX applications, if only to see whether you wish to install a current version. If you have installed all of texlive, the following example will make one of the Arkandis Open Type fonts

available to other applications, and all three of the ParaType TrueType fonts. Adjust or repeat the lines as desired, to either make all the `opentype/` or `truetype/` fonts available, or to select different font directories. As the `root` user:

```
cat > /etc/fonts/conf.d/09-texlive.conf << "EOF"
<?xml version='1.0'?>
<!DOCTYPE fontconfig SYSTEM 'fonts.dtd'>
<fontconfig>
  <dir>/opt/texlive/2024/texmf-dist/fonts/opentype/arkandis/berenisadf</dir>
  <dir>/opt/texlive/2024/texmf-dist/fonts/truetype/paratype</dir>
</fontconfig>
EOF
```

If you do this, remember to change all instances of the year in that file when you upgrade texlive to a later release.

Preferring certain fonts

With the exception of web pages which use WOFF fonts and either supply them or link to google to download them, web pages have traditionally suggested a list of preferred font family names if they cared (e.g. Times New Roman, Serif). There are many reasons why people may wish to have pages which specify a preferred font use a different font, or prefer specific fonts in Monospace or Sans or Serif. As you will expect, there a number of different ways of achieving this.

Fontconfig user documentation

Fontconfig installs user documentation that includes an example 'User configuration file' which among other things prefers [WenQuanYi ZenHei](#) (a Sans font) if a *Serif* font is requested for Chinese (this part might be anachronistic unless you have non-free Chinese fonts, because in `65-nonlatin.conf` this font is already among the preferred fonts when *Serif* is specified for Chinese) and to prefer the modern [VL Gothic](#) font if a Sans font is specified on a Japanese page (otherwise a couple of other fonts would be preferred if they have been installed).

If you have installed the current version, the user documentation is available in HTML, PDF, and text versions at `/usr/share/doc/fontconfig-2.15.0/` : change the version if you installed a different one.

Prefer a specific font

As an example, if for some reason you wished to use the [Nimbus Roman No9 L](#) font wherever Times New Roman is referenced (it is metrically similar, and preferred for Times Roman, but the Serif font from [Liberation fonts](#) will be preferred for the Times New Roman font if installed), as an individual user you could install the font and then create the following file:

```
mkdir -pv ~/.config/fontconfig/conf.d &&
cat > ~/.config/fontconfig/conf.d/35-prefer-nimbus-for-timesnew.conf << "EOF"
<?xml version='1.0'?>
<!DOCTYPE fontconfig SYSTEM 'fonts.dtd'>
<fontconfig>
<!-- prefer Nimbus Roman No9 L for Times New Roman as well as for Times,
without this Tinos and Liberation Serif take precedence for Times New Roman
before Fontconfig falls back to whatever matches Times --&gt;
  &lt;alias binding="same"&gt;
    &lt;family&gt;Times New Roman&lt;/family&gt;
    &lt;accept&gt;
      &lt;family&gt;Nimbus Roman No9 L&lt;/family&gt;
    &lt;/accept&gt;
  &lt;/alias&gt;
&lt;/fontconfig&gt;
EOF</pre>
```

This is something you would normally do in an individual user's settings, but the file in this case has been prefixed '35-' so that it could, if desired, be used system-wide in `/etc/fonts/conf.d/`.

Prefer chosen CJK fonts

The following example of a local configuration (i.e. one that applies for all users of the machine) does several things. It is particularly appropriate where no language is specified, or for reading CJK text in a non-CJK locale, and where the Japanese forms of the codepoints shared with Chinese are preferred. In particular, alternative approaches would be to specify a Chinese font ahead of the Japanese font, meaning that only Kana symbols will be used from the Japanese font, or to not specify DejaVu so that the first font in each set of preferences is preferred for text using Latin alphabets.

1. If a *Serif* font is specified, it prefers [Dejavu fonts](#). If Han codepoints are found, or the Japanese language is specified, the Mincho font from [IPAEx fonts](#) will be used. If Hangul codepoints are found or the Korean language is specified, UnBatang (see [Korean fonts](#)) will be used: Change that line If you installed a different Korean serif font. After that, [WenQuanYi ZenHei](#) (Sans, but a default for *Serif* and monospace) is used. A previous version of this page mentioned using Uming which is a Traditional Chinese font that ships with an old conf file preferring it for zh-tw and zh-hk

language codes (and for sans-serif and monospace). But without the conf file, Fontconfig will only treat it as suitable for zh-hk. The conf file needs to be edited to current style and will then be prepended, so specifying Uming does not belong in this `local.conf` file.

2. For Sans Serif preferences again start with [Dejavu fonts](#), then [VL Gothic](#) for Japanese before falling back to WenQuanYi Zen Hei which is Sans and covers both Chinese and Korean Hangul.
3. The Monospace fonts are forced to the preferred Sans fonts. If the text is in Chinese or Korean then [WenQuanYi ZenHei](#) will be used.

In a non-CJK locale, the result is that suitable fonts will be used for all variants of Chinese, Japanese and Hangul Korean (but Japanese variants of the glyphs shared with Chinese Han will be used). All other languages should already work if a font is present. As the `root` user:

```
cat > /etc/fonts/local.conf << "EOF"
<?xml version='1.0'?>
<!DOCTYPE fontconfig SYSTEM 'fonts.dtd'>
<fontconfig>
    <alias>
        <family>serif</family>
        <prefer>
            <family>DejaVu Serif</family>
            <family>IPAexMincho</family>
            <!-- WenQuanYi is preferred as Serif in 65-nonlatin.conf,
            override that so a real Korean font can be used for Serif -->
            <family>UnBatang</family>
        </prefer>
    </alias>
    <alias>
        <family>sans-serif</family>
        <prefer>
            <family>DejaVu Sans</family>
            <family>VL Gothic</family>
            <!-- This assumes WenQuanYi is good enough for Korean Sans -->
        </prefer>
    </alias>
    <alias>
        <family>monospace</family>
        <prefer>
            <family>DejaVu Sans Mono</family>
            <family>VL Gothic</family>
            <!-- This assumes WenQuanYi is good enough for Korean Monospace -->
        </prefer>
    </alias>
</fontconfig>
EOF
```

Editing Old-Style conf files

Some fonts, particularly Chinese fonts, ship with conf files which can be installed in `/etc/fonts/conf.d`. However, if you do that and then use a terminal to run any command which uses Fontconfig you may see error messages such as :

```
Fontconfig warning: "/etc/fonts/conf.d/69-odofonts.conf", line 14: Having multiple <family> in <alias> isn't supported and may
not work as expected.
```

In practice, these old rules do not work. For non-CJK users, Fontconfig will usually do a good job *without* these rules. Their origin dates back to when CJK users needed handcrafted bitmaps to be legible at small sizes, and those looked ugly next to antialiased Latin glyphs - they preferred to use the same CJK font for the Latin glyphs. There is a side-effect of doing this : the (Serif) font is often also used for Sans, and in such a situation the (English) text in Gtk menus will use this font - compared to system fonts, as well as being serif it is both faint and rather small. That can make it uncomfortable to read.

Nevertheless, these old conf files can be fixed if you wish to use them. The following example is the first part of `64-arpthic-uming.conf` from [Uming](#) - there are many more similar items which also need changing :

```
<match target="pattern">
    <test qual="any" name="lang" compare="contains">
        <string>zh-cn</string>
        <string>zh-sg</string>
    </test>
    <test qual="any" name="family">
        <string>serif</string>
    </test>
    <edit name="family" mode="prepend" binding="strong">
        <string>AR PL Uming CN</string>
```

```
</edit>
</match>
```

The process to correct this is straightforward but tedious - for every item which produces an error message, using your editor (as the `root` user), edit the installed file to repeat the whole block as many times as there are multiple variables, then reduce each example to have only one of them. You may wish to work on one error at a time, save the file after each fix, and from a separate terminal run a command such as `fc-list 2>&1 | less` to see that the fix worked. For the block above, the fixed version will be :

```
<match target="pattern">
    <test qual="any" name="lang" compare="contains">
        <string>zh-cn</string>
    </test>
    <test qual="any" name="family">
        <string>serif</string>
    </test>
    <edit name="family" mode="prepend" binding="strong">
        <string>AR PL UMMing CN</string>
    </edit>
</match>
<match target="pattern">
    <test qual="any" name="lang" compare="contains">
        <string>zh-sq</string>
    </test>
    <test qual="any" name="family">
        <string>serif</string>
    </test>
    <edit name="family" mode="prepend" binding="strong">
        <string>AR PL UMMing CN</string>
    </edit>
</match>
```

About font weights

When this page and the next page were first created, Latin fonts came with a maximum of two weights - either Regular or Book (Book typically has a larger X-height to make it easier to read in large blocks of text), and Bold - and perhaps an Italic (or Slant) style. A few fonts also had Condensed faces (to fit more text into a line and usually only used when specified). Without CSS (Cascading Style Sheets) markup, text used the Regular or Book weight except when ` ... ` markup was used for bold text. Italic styles would be invoked by `<i> ... </i>` markup, along with the bold markup for Bold Italic.

Some faces now contain up to 9 weights, possibly also with a variable font (to save space by including all the alternatives in one file and possibly allowing intermediate weights). For most desktop users who do not need this wide range of weights for creating content, it is simpler to only install one or two weights. If a face has individual weights plus a variable font, the variable font is usually in the top level of the supplied directory, with individual weights in a `static/` subdirectory. Except when initially reviewing a font, it makes no sense to install both static and variable, nor all the possible weights.

The weights are labelled from 100 (Thin) to 900 (Black or Heavy) in CSS terminology, with 400 being normal and 700 bold. The full set of weights is described at [Table of CSS font weights](#).

If you have installed a font with a range of weights, you can copy [font-weights.html](#) to your local machine. As shipped it will use your default Serif font assuming you have one. Edit it to point to a specific installed font using the name known to Fontconfig (also in the `*EDITME FONTNAME*` text items) and open it from your desktop browser. You can also use it to look at a font with only two installed weights, e.g. for testing to see if you prefer other weights.

Despite the details in that Mozilla link, it appears that if only normal and bold weights are installed, SemiBold (600) will be shown using bold.

There seems to be a little scope for changing which weights are used for normal and bold *if only two weights have been installed*. Firefox, and probably other browsers, will look for the next weight heavier than normal. If that is less than bold (Medium, maybe SemiBold - uncertain) it will be used for normal and then the next higher weight, if any will be used for bold, allowing you to make the fonts slightly darker. Conversely, if only a weight less than normal has been installed, such as Light, that will be used for both normal and bold weights (the upward search happens first).

If you remove some weights of a system font, you may need to run `fc-cache` as the `root` user and then log out completely to clear caches associated with your user.

Items which can override Fontconfig

Several desktop environments, as well as some programs, will use Fontconfig to find fonts but may override certain things.

GNOME: The settings in `org.gnome.desktop.interface` can be updated with dconf-editor. You can set the fonts to your preference and desired point size. To use the fonts chosen by Fontconfig specify e.g. 'Sans 12', 'Serif 11', 'Mono 10' as desired. Also review the antialiasing, hinting and rgba settings. Alternatively, [gnome-tweaks-46.1](#) can also update the font settings in a GUI form.

LXQt: Change font settings as necessary to match Fontconfig in lxqt-config-appearance.

KDE Plasma: The settings can be adjusted in System Settings under Appearance -> Fonts or in Edit -> Preferences of the individual application.

Xfce desktop: The settings can be adjusted in Settings -> Appearance -> Fonts. Specify your preferred fonts, e.g. 'Sans Regular' (to use the normal face and weight rather than Bold and/or Italic) and adjust the point size in the option. Review the Rendering and DPI options.

Firefox: This browser allows you to specify its default fonts. For the 128esr series use the 'Hamburger' menu to go to Settings, General, and under Fonts -> Advanced select Sans Serif, Serif or Monospace as appropriate if you wish to use the fonts which match Fontconfig. Set the point sizes as desired. In later versions, the settings are at Preferences -> Fonts.

Libreoffice: Tests using English text with an old Japanese font (HanaMinA) which supports several European languages but had only one weight and no italics or slant showed that although Libreoffice uses Fontconfig to find the font, it created its own bold or slanted text. It is not clear if it will do the same where a font actually has bold weight or an italic style. Also, documentation shows that Libreoffice has its own substitution rules for when a codepoint is not found in the selected font, but is unclear if those rules apply on Linux using Fontconfig.

Pango: as noted in the example `~/.config/fontconfig` above, anything using Pango-1.44 (from 2019) or later now uses Harfbuzz for hinting, not FreeType, and `hintfull` is not supported.

Thunderbird: The font settings can be changed by going to "Edit -> Settings" and then scrolling down to "Fonts & Colors".

External Links

I stared into the fontconfig ...

The blog entries by [Fevée](#) are particularly useful if Fontconfig does not think your chosen font supports your language, and for preferring some non-MS Japanese fonts when an ugly MS font is already installed.

subpixel-hinting

The documentation of the FreeType v40 interpreter at [freetype docs](#) explains how the current hinter works, and why the previous (slow) Infinality interpreter was replaced.

Calculating DPI

An old answer at [askubuntu](#) gives some detail on calculating a screen's dots per inch, but essentially you just measure the width and height of the visible panel, convert to inches if using metric measurements, and divide by the number of pixels. You can then pass `-dpi 90` when you start Xorg, using your own value.

Table of CSS font weights

Perhaps more than you ever wished to know is at [Mozilla CSS docs](#).

Applying autohinting to a font

If you are using hinting and have a TTF (not OTF) font which lacks hints but permits you to fork it, you might be able to apply hints using [ttfautohint](#) which is based on the old autohinter. As of version 1.8.4 it fails to build without Qt5.

Fontconfig in the Arch wiki

Arch has a lot of information in its wiki at [font configuration](#).

Fontconfig in the Gentoo wiki

Gentoo has some information in its wiki at [Fontconfig](#) although a lot of the details (what to enable, and Infinality) are specific to Gentoo.

TTF and OTF fonts

About TTF and OTF fonts

Originally, Xorg provided only bitmap fonts. Later, some scalable Type1 fonts were added, but the desktop world moved on to using TrueType and Open Type fonts. To support these, Xorg uses Xft, the X FreeType interface library, with Fontconfig (see previous page for details including how fonts are selected and various reasons why a font might be ignored).

A few fonts are provided as collections (TTC or OTC) where font data is shared between different fonts, which saves disk space. These should be treated in exactly the same way as individual TTF or OTF files.

If a font provides both TTF and OTF forms, you should prefer the OTF form in Linux, as it may provide more features for programs which know how to use them (such as xelatex). The TTF files of these fonts usually contain hinting, see [Hinting and Anti-aliasing](#) and [Items which can override Fontconfig](#) for reasons why hinting may be undesirable or unusable.

A font may have multiple variations. For example, Noto Sans has 9 weights (ExtraLight, Light, Thin, Normal, Medium, SemiBold, Bold, ExtraBold, and Black) and 2 styles (Regular and Italic), thus 18 variations in total. Normally each variation is provided as a separate TTF or OTF file. For full coverage you need to install all these TTF or OTF files. Even if you are low on disk space, you should still install two weights (Regular and Bold) by two styles (Normal and Italic) if the font has these variations. Some fonts do not have Italic style (for example most CJK fonts and some monospace fonts), and some fonts only have one variation (for example Noto Sans Math, it only provides the glyph of some mathematic symbols).

Some fonts are also available as *variable* font files. Unlike a normal font file which only contains one variation, a variable font file contains infinite variations. Each variation can be defined by the application using this font by assigning number(s) to one or more variables. There are also pre-defined *named instances* analogous to the traditional variations. For example, with the variable version of Noto Sans, the weight variable can be assigned any number not less than 100 and not greater than 900, and 9 named instances are pre-defined: ExtraLight for weight=100, Regular for weight=400, Bold for weight=700, etc. So once a variable font file for Noto Sans is installed, all the 9 named instances (or "variations") are available. Note that the slope is not defined as a variable in the variable version of Noto Sans, so Regular and Italic are still traditional variations and a separate variable font file is needed for the Italic variation.

A variable font file is obviously more flexible than the normal (static) font files. It's extremely useful for fine tuning the font for Web pages or publications. And, the size of a variable font file is usually significantly smaller than the total size of several static font files for multiple variations. For example, the variable font file for Noto Sans SC is only 11M, while the total size of 9 static font files for Noto Sans SC is 91M. But you must make sure your applications really support variable fonts before installing one. For example, `lualatex` supports variable font but `xelatex` does not. So if you want to use a font for an article and use `xelatex` for typesetting, you must not install the variable font files.

For information about variable fonts, please see [Variable Fonts](#).

For some scripts, Pango is required to render things correctly, either by selecting different glyph forms, or by combining glyphs - in both cases, according to the context. This applies particularly to Arabic and Indic scripts.

Standard scalable fonts that come with X provide very poor Unicode coverage. You may notice in applications that use Xft that some characters appear as a box with four binary digits inside. In this case, a font with the required glyphs has not been found. Other times, applications that don't use other font families by default and don't accept substitutions from Fontconfig will display blank lines when the default font doesn't cover the orthography of the user's language.

The fonts available to a program are those which were present when it was started, so if you add an extra font and wish to use it in a program which is currently running, then you will have to close and restart that program.

Some people are happy to have dozens, or even hundreds, of font files available, but if you ever wish to select a specific font in a desktop application (for example in a word processor) then scrolling through a lot of fonts to find the right one is slow and awkward - fewer is better. So, for some font packages you might decide to install only one of the fonts - but nevertheless install the different variants (italic, bold, etc) as these are all variations for the same font name.

In the past, everybody recommended running `fc-cache` as the `root` user after installing or removing fonts, but this is not necessary anymore on Linux, Fontconfig will do it automatically if needed as well as if the font caches are more than 30 seconds old. However, if you add a font and want to use it immediately, you can run that command as a normal user.

There are several references below to CJK characters. This stands for Chinese, Japanese and Korean, although modern Korean is now almost all written using the phonetic Hangul glyphs (it used to sometimes use Hanja glyphs which are similar to Chinese and Japanese). Unicode decided to go for [Han Unification](#) and to map some Chinese and Japanese glyphs to the same codepoints. This was very unpopular in Japan, and the result is that different fonts will render some codepoints in quite different shapes. In addition, Simplified Chinese will sometimes use the same codepoint as Traditional Chinese but will show it differently, somewhat analogous to the different shapes used for the letters 'a' and 'g' in English (single-storey and two-storey), except that in a language context one will look "wrong" rather than just "different".

Unlike most other packages in this book, the BLFS editors do not monitor the versions of the fonts on this page - once a font is good enough for general use, the typical additions in a new version are minor (e.g. new currency symbols, or glyphs not for a modern language, such as emojis or playing cards). Therefore, none of these fonts show version or md5 information.

The list below will not provide complete Unicode coverage. Unicode is updated every year, and most additions are now for historic writing systems. For almost-complete coverage you can install the full set of [Noto fonts](#) (there are separate fonts for different writing systems). We used to recommend the [Unicode Font Guide](#), but that has not been updated since 2008 and many of its links are dead.

Rendered examples of most of these fonts, and many others, with details of what languages they cover, can be found at a [font-comparison](#) website.

Fonts are often supplied in zip files, requiring [UnZip-6.0](#) to list and extract them, but even if the current release is a tarball, you should still check to see if it will create a directory (scattering the contents of a zipfile or tarball across the current directory can be very messy, and a few fonts create __MACOSX/ directories). In addition, many fonts are supplied with permissions which do not let 'other' users read them - if a font is to be installed for system-wide use, any directories must be mode 755 and all the files mode 644, so you need to change them if the permissions are different. If you forget, the root user may be able to see a particular font in `fc-list`, but a normal user will not be able to use them.

As a font installation example, consider the installation of the [Dejavu fonts](#). In this particular package, the TTF files are in a subdirectory. From the unpacked source directory, run the following commands as the `root` user:

```
install -v -d -m755 /usr/share/fonts/dejavu &&
install -v -m644 ttf/*.ttf /usr/share/fonts/dejavu &&
fc-cache -v /usr/share/fonts/dejavu
```

If you wish, you can also install any licenses or other documentation, either alongside the font or in a corresponding directory under `/usr/share/doc/`.

A few fonts ship with source as well as the completed TTF or OTF file(s). Unless you intend to modify the font, and have the correct tools (sometimes [FontForge-20230101](#), but often commercial tools), the source will provide no benefit, so do not install it. One or two fonts even ship with Web Open Font Format (WOFF) files - this is useful if you run a webserver and want to use that font on your website, but not useful for a desktop system.

To provide greater Unicode coverage, you should install some of the following fonts, depending on what websites and languages you want to read. The next part of this page details some fonts which cover at least Latin alphabets, and the final part deals with some CJK issues.

Note

Installation of the [Dejavu fonts](#) is strongly recommended.

Caladea

[Caladea](#) (created as a Chrome OS extra font) is metrically compatible with MS Cambria and can be used if you have to edit a document which somebody started in Microsoft Office using Cambria.

Cantarell fonts

[Cantarell fonts](#) – The Cantarell typeface family provides a contemporary Humanist Sans Serif. It is particularly optimised for legibility at small sizes and is the preferred font family for the GNOME user interface.

Carlito

[Carlito](#) (created as another Chrome OS extra font) is metrically compatible with MS Calibri and can be used if you have to edit a document which somebody started in Microsoft Office using Calibri.

DejaVu fonts

[DejaVu fonts](#) – These fonts are an extension of, and replacement for, the Bitstream Vera fonts and provide Latin-based scripts with accents and punctuation such as "smart-quotes" and variant spacing characters, as well as Cyrillic, Greek, Arabic, Hebrew, Armenian, Georgian and some other glyphs. In the absence of the Bitstream Vera fonts (which had much less coverage), these were the default fallback fonts for 'Latin' languages. As of Fontconfig-2.14, if the Noto 'Latin' fonts have been installed they will be prioritised ahead of DejaVu.

GNU FreeFont

[GNU FreeFont](#) – This set of fonts covers many non-CJK characters, but the glyphs are comparatively small (unlike Noto and DejaVu fonts which are comparatively large) and rather light weight ("less black" when black on white is used) which means that in some contexts such as terminals they are not visually pleasing, for example when most other glyphs are provided by another font. On the other hand, some fonts used primarily for printed output, and many CJK fonts, are also light weight.

Gelasio

[Gelasio](#) is metrically compatible with MS Georgia and Fontconfig will use it if MS Georgia is requested but is not installed.

Liberation fonts

The [Liberation fonts](#) provide libre substitutes for Arial, Courier New, and Times New Roman. Fontconfig will use them as substitutes for those fonts, and also for the similar Helvetica, Courier, and Times Roman, though for these it can prefer a different font (see the examples in the PDFs at [typoetting.co.uk](#)).

Many people will find the Liberation fonts useful for pages where one of those fonts is requested.

Microsoft Core Fonts

The [Microsoft Core Fonts](#) date from 2002. They were supplied with old versions of Microsoft Windows and were apparently made available for general use. You can extract them from the '.exe' files using bsd-tar from [libarchive-3.7.4](#).

Make sure that you read the license before using them. At one time some of these fonts (particularly Arial, Times New Roman, and to a lesser extent Courier New) were widely used on web pages. The full set contains Andale Mono, Arial, Arial Black, Comic Sans MS, Courier New, Georgia, Impact, Times New Roman, Trebuchet MS, Verdana and Webdings.

Please note that if you only want to use a font with the same metrics (character size, etc) as Arial, Courier New, or Times New Roman you can use the libre Liberation Fonts (above), and similarly you can replace Georgia with Gelasio.

Although many old posts recommend installing these fonts for output which looks better, less old posts say that these are 'ugly' or 'broken' with modern Fontconfig, Freetype and Pango. Most people will not want to install any of these fonts.

The newer fonts which Microsoft made their defaults in later releases of MS Windows or MS Office (Calibri and Cambria) have never been freely available. However, if you do not have them installed you can find metric equivalents (Carlito and Caladea) above.

Noto fonts

The [Noto fonts](#) ('No Tofu', i.e. avoiding boxes with dots [hex digits] when a glyph cannot be found) are a set of fonts which aim to cover every *glyph in Unicode, no matter how obscure*.

People using languages written in Latin, Greek or Cyrillic alphabets only need to install Noto Sans, Noto Serif and/or Noto Sans Mono. For more details on the organization of Noto fonts see [how are noto fonts organized](#). There are also separate fonts for every other current writing system, but those do not cover text in Latin languages.

Those three fonts are the first-choice preferences of Fontconfig for text in those three alphabets, and they are updated frequently to cover recent Unicode additions such as the extensions for phonetic transcription in Latin and Cyrillic alphabets.

It may be easier to download a specific Noto font by going to [Noto Sans](#) and changing the font name as appropriate, with '+' between each word, e.g. 'Noto+Kufi+Arabic', 'Noto+Serif+Georgian' or whatever, then clicking on 'Download family'.

For the Noto CJK fonts it is easier to find the specific zip archive you desire at Github. Go to <https://github.com/notofonts/noto-cjk/releases/> and look for the newest Sans or Serif version with its own updated Download guide and its assets. The Download guides should help you to identify which zip file best matches your requirements.

Noto Sans is the preferred font for KDE Plasma and applications, except for monospace fonts where [Hack](#) is preferred.

For writing systems not using the Latin, Greek or Cyrillic alphabets the Noto fonts are not preferred by Fontconfig. If you use a Noto font for a modern language where another installed font also covers it (e.g. the DejaVu fonts cover several Right-to-Left alphabets), you might need to set a preference for Fontconfig - see the previous page.

Source Code Pro

This set of fonts from Adobe (seven different weights) includes what is now the preferred monospace font for those applications which use [gsettings-desktop-schemas-46.1](#). The github release [source-code-pro](#) contains OTF (preferred) and TTF as well as the source and WOFF fonts.

To use this in terminals, you probably will only want the Regular font.

There is also an older TTF version of this available from [Google fonts](#) but that has very limited coverage (adequate for most European languages using a Latin alphabet).

Lohit fonts

For Indic languages, Fontconfig now prefers Lohit fonts (Sanskrit for 'red'). They can be found at [pagure.org](#)

CJK fonts

As indicated earlier, usage of a combination of Chinese, Japanese and Korean characters can be tricky - each font only covers a subset of the available codepoints, the preferred shapes of the glyphs can differ between the languages, and many of the CJK fonts do not actually support modern Korean.

Also, Fontconfig prefers Chinese to Japanese by default. Tuning that is covered at [Preferring chosen CJK fonts](#).

Although Unicode has been extended to allow a very large number of CJK codepoints, those outside the Base Plane (greater than U+0xFFFF) are not commonly used in Mandarin (the normal form of written Chinese, whether Simplified (Mainland China, Malaysia, and Singapore) or Traditional (Hong Kong and Taiwan)), or Japanese.

For Hong Kong, which uses Traditional Chinese and where Cantonese is the dominant language, the Hong Kong Supplementary Character Set was added to Unicode in 2005 and revised in 2009 (it is part of CJK Extension B and contains more than 1900 characters). Earlier fonts will not be able to support either Cantonese or use these characters where local names are written in Mandarin. The Uming HK, Noto Sans HK and WenQuanYi Zen Hei fonts all seem to cover Hong Kong usage (Fontconfig disagrees about Noto Sans HK).

The Han glyphs are double width, and other glyphs in the same font may be narrower. For their CJK content, all of these fonts can be regarded as monospaced (i.e. fixed width).

If you wish to use Noto fonts, there are also Serif versions of their various CJK fonts. The Noto Sans/Serif SC/TC/HK/JP/KR fonts are derived from a monolithic [noto-cjk](#) repository and you can find the `.ttc` files for the entire Noto Sans CJK (including SC/TC/HK/JP/KR) or Noto Serif CJK font family there. Google recommends the normal users to use the separate Noto Sans/Serif SC/TC/HK/JP/KR fonts instead, but if you are capable and willing to read texts in more than one CJK character systems it may be easier to use a monolithic `.ttc` file for full coverage.

If all you wish to do is render CJK glyphs, installing [WenQuanYi ZenHei](#) may be a good place to start if you do not already have a preference.

Chinese fonts:

In Chinese, there are three font styles in common use: Sung (also known as Song or Ming), which is the most-common ornamented ("serif") form, Kai ("brush strokes") which is an earlier ornamented style that looks quite different, and modern Hei ("sans"). Unless you appreciate the differences, you probably do not want to install Kai fonts.

The current versions of Chinese Noto Sans fonts can be found at [Noto Sans SC](#) for Simplified Chinese, [Noto Sans TC](#) for Traditional Chinese, and as mentioned above [Noto Sans HK](#) for use in Hong Kong.

Opendesktop fonts

A copy of version 1.4.2 of the [opendesktop-fonts](#) is preserved at Arch. This was a later development of fireflysung which BLFS used to recommend, adding Kai and Mono fonts. The name of the Sung font remains 'AR PL New Sung' so they cannot both be installed together.

At one time there was a 1.6 release, and more recently some versions at github, which also included a Sans font (Odohei), but those have dropped off the web and it is unclear if there was a problem. Fontconfig does not know anything about the later fonts (AR PL New Kai, AR PL New Sung Mono) and will default to treating them as Sans.

UMing

[UMing fonts](#) – sets of Chinese Ming fonts (from Debian, use the '.orig' tarball) in a ttc which contain variations of Simplified and Traditional Chinese (Taiwanese, with second variant for different [bopomofo](#), and Cantonese for Hong Kong). This ships with old-syntax files which you can install to `/etc/fonts/conf.d/` but see [Editing Old-Style conf files](#).

WenQuanYi Zen Hei

[WenQuanYi Zen Hei](#) provides a Sans-Serif font which covers all CJK scripts including Korean. Although it includes old-style conf files, these are not required: Fontconfig will already treat these fonts (the 'sharp' contains bitmaps, the monospace appears not to be Mono in its ASCII part) as Sans, Serif, and Monospace. If all you wish to do is to be able to render Han and Korean text without worrying about the niceties of the shapes used, the main font from this package is a good font to use.

Japanese fonts:

In Japanese, Gothic fonts are Sans, and Mincho are Serif. BLFS used to only mention the Kochi fonts, but those appear to now be the least-preferred of the Japanese fonts.

Apart from the fonts detailed below, also consider [Noto Sans JP](#).

IPAex fonts

The [IPAex fonts](#) are the current version of the IPA fonts. Use [Google Translate](#) on the home page, then click on the download link for IPAex Font Ver.004.01. Unfortunately, Fontconfig only knows about the older IPAfonts and the forked IPA Mona font (which is not easily available and which apparently does not meet Debian's Free Software guidelines). If you install the IPAex fonts, you may want to make it known to Fontconfig. Please see [Preferring chosen CJK fonts](#) for one way to accomplish this.

Kochi fonts

The [Kochi Substitute fonts](#) were the first truly libre Japanese fonts (the earlier Kochi fonts were allegedly plagiarized from a commercial font).

VL Gothic

The [VL Gothic](#) font is a modern Japanese font in two variants with monotonic or proportional spacing for the non-Japanese characters.

Korean fonts:

In Korean, Batang or Myeongjo (the older name) are Serif, Dotum or Gothic are the main Sans fonts. BLFS previously recommended the Baekmuk fonts, but the Nanum and Un fonts are now preferred to Baekmuk by Fontconfig because of user requests.

A convenient place to see examples of these and many other Korean fonts is [Free Korean Fonts](#). Click on 'Gothic Fonts' or 'All Categories -> Myeongjo Fonts', then click on the font example to see more details including the License, and click on the link to download it. For Nanum, you will need to be able to read Korean to find the download link on the page you get to. For Un there are direct links and you can find the un-fonts-core tarball in the `releases/` directory.

Alternatively, consider [Noto Sans KR](#) or [WenQuanYi ZenHei](#).

Xorg Legacy

Introduction to Xorg Legacy

Xorg's ancestor (X11R1, in 1987) at first only provided bitmap fonts, with a tool (`bdf2pcf`) to assist in their installation. With the introduction of `xorg-server-1.19.0` and `libXfont2` many people will not need them. There are still a few old packages which might require, or benefit from, these deprecated fonts and so the following packages are shown here.

Note

The `font-adobe-100dpi` package installs 100 dots per inch versions of Courier, Helvetica, New Century Schoolbook and Times fonts.

The remaining font packages here provide 75dpi versions of those fonts, and various miscellaneous fonts - mostly for legacy encodings. Those are primarily to silence warnings when testing Xorg. In previous versions of BLFS a lot more legacy fonts were installed.

Please consult the BLFS 7.10 `systemd` book at <https://www.linuxfromscratch.org/blfs/view/7.10-systemd/x/x7font.html> if you wish to install any of those other fonts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.x.org/pub/individual/>
- Download size: 4.2 MB
- Estimated disk space required: 30 MB
- Estimated build time: 0.3 SBU

Xorg Legacy Dependencies

Required

[Xorg Fonts](#)

Downloading Xorg Legacy

First, create a list of files to be downloaded. This file will also be used to verify the integrity of the downloads when complete:

```
cat > legacy.dat << "EOF"
e09b61567abda4d34119bba24eddfb1 util/ bdf2pcf-1.1.1.tar.xz
20239f6f99ac586f10360b0759f73361 font/ font-adobe-100dpi-1.0.4.tar.xz
2dc044f693ee8e0836f718c2699628b9 font/ font-adobe-75dpi-1.0.4.tar.xz
2c939d5bd4609d8e284be9bef4b8b330 font/ font-jis-misc-1.0.4.tar.xz
6300bc99a1e45fbbe6075b3de728c27f font/ font-daewoo-misc-1.0.4.tar.xz
fe2c44307639062d07c6e9f75f4d6a13 font/ font-isas-misc-1.0.4.tar.xz
145128c4b5f7820c974c8c5b9f6ffe94 font/ font-misc-misc-1.1.3.tar.xz
EOF
```

To download the needed files using [Wget-1.24.5](#), use the following commands:

```
mkdir legacy &&
cd legacy &&
grep -v '^#' ./legacy.dat | awk '{print $2$3}' | wget -i- -c \
-B https://www.x.org/pub/individual/ &&
grep -v '^#' ./legacy.dat | awk '{print $1 " " $3}' > ./legacy.md5 &&
md5sum -c ./legacy.md5
```

Installation of Xorg Legacy

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

1. Run the entire script as the root user (not recommended).
2. Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
3. Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```
as_root()
{
    if [ $EUID = 0 ];      then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\\"
    fi
}

export -f as_root
```

First, start a subshell that will exit on error:

```
bash -e
```

Install all of the packages by running the following commands:

```
for package in $(grep -v '^#' ../*.md5 | awk '{print $2}')
do
    packagedir=${package%.*}
    tar -xf $package
    pushd $packagedir
    ./configure $XORG_CONFIG
    make
    as_root make install
    popd
    rm -rf $packagedir
    as_root /sbin/ldconfig
done
```

Finally, exit the shell that was started earlier:

```
exit
```

Contents

Installed Programs: `bdftopcf`

Installed Libraries: None

Installed Directories: `$XORG_PREFIX/share/fonts/100dpi`, `$XORG_PREFIX/share/fonts/75dpi`, `$XORG_PREFIX/share/fonts/misc`

Short Descriptions

`bdftopcf` converts an X font from Bitmap Distribution Format to Portable Compiled Format

Chapter 25. Graphical Environment Libraries

This chapter does not contain libraries that are required to run X. It does contain libraries that enhance X. In some cases the enhancement is as simple as font support. In others it is as complex as libraries that sit between X and applications that run on X whose purpose is to standardize the look and feel and inter-process communications for different applications. They also assist programmers by supplying common elements.

Introduction to Atkmm

Atkmm is the official C++ interface for the ATK accessibility toolkit library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/atkmm/2.28/atkmm-2.28.4.tar.xz>
- Download MD5 sum: e0f7271990c89a6c9987f215ba47bfc5
- Download size: 724 KB
- Estimated disk space required: 14 MB
- Estimated build time: 0.2 SBU

Atkmm Dependencies

Required

[at-spi2-core-2.52.0](#) and [GLibmm-2.66.7](#)

Installation of Atkmm

Install Atkmm by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Library: libatkmm-1.6.so

Installed Directories: /usr/{include,lib}/atkmm-1.6 and /usr/share/{devhelp/books/atkmm-1.6,doc/atkmm-2.28.4}

Short Descriptions

libatkmm-1.6.so contains the ATK API classes

Atkmm-2.36.3

Introduction to Atkmm

Atkmm is the official C++ interface for the ATK accessibility toolkit library. This version is part of a new API for supporting gtkmm-4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/atkmm/2.36/atkmm-2.36.3.tar.xz>
- Download MD5 sum: 8b03a39a5e3dd0d3c040ece345f7a786
- Download size: 780 KB
- Estimated disk space required: 11 MB
- Estimated build time: 0.2 SBU

Atkmm Dependencies

Required

[at-spi2-core-2.52.0](#) and [GLibmm-2.80.0](#)

Installation of Atkmm

Install Atkmm by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Library: libatkmm-2.36.so

Installed Directories: /usr/{include,lib}/atkmm-2.36 and /usr/share/{devhelp/books/atkmm-2.36,doc/atkmm-2.28.4}

Short Descriptions

libatkmm-2.36.so contains the ATK API classes

at-spi2-core-2.52.0

Introduction to At-Spi2 Core

The At-Spi2 Core package contains a comprehensive accessibility framework for the Assistive Technologies available on the GNOME platform. This includes a set of interfaces which are implemented by other toolkits and applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/at-spi2-core/2.52/at-spi2-core-2.52.0.tar.xz>
- Download MD5 sum: e6591545b2bf204fe9a58f777bd0b78a
- Download size: 564 KB
- Estimated disk space required: 15 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

At-Spi2 Core Dependencies

Required

[dbus-1.14.10](#), [GLib-2.80.4](#) (GObject Introspection required for GNOME), [gsettings-desktop-schemas-46.1](#) (Runtime), and [Xorg Libraries](#)

Optional

[Gi-DocGen-2024.1](#) and [sphinx-8.0.2](#)

Installation of At-Spi2 Core

Install At-Spi2 Core by running the following commands:

```
mkdir build &&
cd      build &&
```

```
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

Now, as the `root` user:

```
ninja install
```

The test suite requires the glib schemas of the package to be installed already. The tests also need to run in a graphical environment. Now that the package is installed, test the results by issuing: `dbus-run-session ninja test`.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D docs=true`: Use this switch if you want to build the documentation. Note that you must have both [Gi-DocGen-2024.1](#) and [sphinx-8.0.2](#) installed on your system.

Contents

Installed Programs: None

Installed Library: libatk-1.0.so, libatk-bridge-2.0.so, libatspi.so, and /usr/lib/gtk-2.0/modules/libatk-bridge.so

Installed Directories: /usr/include/atk-1.0, /usr/include/at-spi-2.0, /usr/include/at-spi2-atk, /usr/lib/gnome-settings-daemon-3.0, /usr/share/defaults/at-spi2, and /usr/share/gtk-doc/html/libatspi (optional)

Short Descriptions

libatk-1.0.so	contains functions that are used by assistive technologies to interact with desktop applications
libatk-bridge.so	contains the Accessibility Toolkit GTK+-2 bridge
libatk-bridge-2.0.so	contains the Accessibility Toolkit GTK+ module
libatspi.so	contains the At-Spi2 API functions

Cairo-1.18.0

Introduction to Cairo

Cairo is a 2D graphics library with support for multiple output devices. Currently supported output targets include the X Window System, Win32, image buffers, PostScript, PDF and SVG. Experimental backends include OpenGL, Quartz and XCB file output. Cairo is designed to produce consistent output on all output media while taking advantage of display hardware acceleration when available (e.g. through the X Render Extension). The Cairo API provides operations similar to the drawing operators of PostScript and PDF. Operations in Cairo include stroking and filling cubic Bézier splines, transforming and compositing translucent images, and antialiased text rendering. All drawing operations can be transformed by any [affine transformation](#) (scale, rotation, shear, etc.)

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cairographics.org/releases/cairo-1.18.0.tar.xz>
- Download MD5 sum: 3f0685fbadc530606f965b9645bb51d9
- Download size: 32 MB
- Estimated disk space required: 98 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

Cairo Dependencies

Required

[libpng-1.6.43](#) and [Pixman-0.43.4](#)

Recommended

[Fontconfig-2.15.0](#), [GLib-2.80.4](#) (required for most GUIs), and [Xorg Libraries](#)

Optional

[ghostscript-10.03.1](#), [GTK+-3.24.43](#), [GTK-Doc-1.34.0](#), [libdrm-2.4.122](#), [librsvg-2.58.3](#), [libxml2-2.13.3](#), [LZO-2.10](#), [Mesa-24.1.5](#), [Poppler-24.08.0](#), [Valgrind-3.23.0](#), [GTK+-2](#), [jbig2dec](#), [libspectre](#), and [Skia](#)

Note

There is a circular dependency between cairo and harfbuzz. If cairo is built before harfbuzz, it is necessary to rebuild cairo after harfbuzz in order to build pango.

Installation of Cairo

Install Cairo by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not have a working test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D xlib-xcb(enabled)`: This switch enables several experimental Xlib/XCB functions used by some window managers.

`-D gtk_doc(true)`: Use this parameter if GTK-Doc is installed and you wish to create and install the documentation.

Contents

Installed Programs: cairo-trace

Installed Libraries: libcairo.so, libcairo-gobject.so, and libcairo-script-interpreter.so

Installed Directories: /usr/include/cairo and /usr/lib/cairo

Short Descriptions

<code>cairo-trace</code>	generates a log of all calls made by an application to Cairo
<code>libcairo.so</code>	contains the 2D graphics functions required for rendering to the various output targets
<code>libcairo-gobject.so</code>	contains functions that integrate Cairo with Glib's GObject type system
<code>libcairo-script-interpreter.so</code>	contains the script interpreter functions for executing and manipulating Cairo execution traces

libcairomm-1.0 (cairomm-1.14.5)

Introduction to libcairomm-1.0

The libcairomm-1.0 package provides a C++ interface to Cairo.

This package is known to build and work properly using LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cairographics.org/releases/cairomm-1.14.5.tar.xz>
- Download MD5 sum: 0974ef291d491f22df287f588580677d
- Download size: 638 KB

- Estimated disk space required: 9.5 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

libcairomm-1.0 Dependencies

Required

[Cairo-1.18.0](#) and [libsigc++-2.12.1](#)

Recommended

[Boost-1.86.0](#) (for tests)

Optional

[Doxygen-1.12.0](#)

Installation of libcairomm-1.0

Install Cairomm-1.0 by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D build-tests=true \
    -D boost-shared=true &&
ninja
```

To run the test suite, run: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D build-tests=true`: This switch is for building the unit tests. Remove if you have not installed [Boost-1.86.0](#).

`-D boost-shared=true`: This switch has the package use the shared version of boost libraries. It is required if you have not installed the boost static libraries, and you have passed `-D build-tests=true`.

`-D build-documentation=true`: This switch builds the html documentation if doxygen is installed.

Contents

Installed Programs: None

Installed Library: libcairomm-1.0.so

Installed Directories: /usr/{lib,include}/cairomm-1.0 and /usr/share/{devhelp/books,doc}/cairomm-1.0 (optional)

Short Descriptions

libcairomm-1.0.so contains the Cairo API classes

libcairomm-1.16 (cairomm-1.18.0)

Introduction to libcairomm-1.16

The libcairomm-1.16 package provides a C++ interface to Cairo. This version of the API is needed to support gtkmm-4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.cairographics.org/releases/cairomm-1.18.0.tar.xz>
- Download MD5 sum: 4c7afc4ab5177655724ea4b31794db30
- Download size: 620 KB
- Estimated disk space required: 25 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libcairomm-1.16 Dependencies

Required

[Cairo-1.18.0](#) and [libsigc++-3.6.0](#)

Recommended

[Boost-1.86.0](#) (for tests)

Optional

[Doxygen-1.12.0](#)

Installation of libcairomm-1.16

Install Cairomm-1.16 by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D build-tests=true \
    -D boost-shared=true &&
ninja
```

To run the test suite, run: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D build-tests=true`: This switch is for building the unit tests. Remove if you have not installed [Boost-1.86.0](#).

`-D boost-shared=true`: This switch has the package use the shared version of boost libraries. It is required if you have not installed the boost static libraries, and you have passed `-D build-tests=true`.

`-D build-documentation=true`: This switch builds the html documentation if doxygen is installed.

Contents

Installed Programs: None

Installed Library: libcairomm-1.16.so

Installed Directories: /usr/{lib,include}/cairomm-1.16 and /usr/share/{devhelp/books,doc}/cairomm-1.16 (optional)

Short Descriptions

libcairomm-1.16.so contains the Cairo API classes

colord-gtk-0.3.1

Introduction to Colord GTK

The Colord GTK package contains GTK+ bindings for Colord.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/colord/releases/colord-gtk-0.3.1.tar.xz>
- Download MD5 sum: d436740c06e42af421384f16b2a9a0a7
- Download size: 24 KB
- Estimated disk space required: 2.1 MB
- Estimated build time: less than 0.1 SBU

Colord GTK Dependencies

Required

[colord-1.4.7](#) and [GTK+-3.24.43](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [GTK-4.14.5](#), and [Vala-0.56.17](#)

Optional

[DocBook-utils-0.6.14](#) and [GTK-Doc-1.34.0](#)

Installation of Colord GTK

Warning

If building the documentation `ninja -j1` must be used.

Install Colord GTK by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk4=true        \
            -D vapi=true         \
            -D docs=false        \
            -D man=false         \
            ..
&&
ninja
```

To test the results, issue: `ninja test`. The tests need to be run from an X session, and may require a color profile for your primary display.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D vapi=true`: This switch enables building of the Vala bindings. Remove this switch if you don't have [Vala-0.56.17](#) installed.

`-D gtk4=true`: This switch enables building the GTK-4 bindings for colord. Set this option to 'false' if you do not have [GTK-4.14.5](#) installed.

`-D docs=false`: This switch disables building gtk-doc based documentation. Even if gtk-doc is installed, you will need the namespaced versions of the Docbook XSL stylesheets.

-D man=false: This switch disables generating the manual pages for this package. Remove this switch if you have namespaced versions of the Docbook XSL stylesheets installed.

Contents

Installed Programs: cd-convert

Installed Libraries: libcolord-gtk.so and libcolord-gtk4.so

Installed Directories: /usr/include/colord-1/colord-gtk and /usr/share/gtk-doc/html/colord-gtk

Short Descriptions

cd-convert	is a Color Manager Testing Tool
libcolord-gtk.so	contains the Colord GTK+ bindings
libcolord-gtk4.so	contains the Colord GTK-4 bindings

FLTK-1.3.9

Introduction to FLTK

FLTK (pronounced "fulltick") is a cross-platform C++ GUI toolkit. FLTK provides modern GUI functionality and supports 3D graphics via OpenGL and its built-in GLUT emulation libraries used for creating graphical user interfaces for applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://fltk.org/pub/fltk/1.3.9/fltk-1.3.9-source.tar.gz>
- Download MD5 sum: 6e7a389f97b420bc31f99edd3522a85e
- Download size: 5.4 MB
- Estimated disk space required: 122 MB (with documentation)
- Estimated build time: 0.2 SBU (Using parallelism=4)

FLTK Dependencies

Required

[Xorg Libraries](#)

Recommended

[hicolor-icon-theme-0.18](#), [libjpeg-turbo-3.0.1](#), and [libpng-1.6.43](#)

Optional

[alsa-lib-1.2.12](#), [desktop-file-utils-0.27](#), [Doxygen-1.12.0](#), [GLU-9.0.3](#), [Mesa-24.1.5](#), and [texlive-20240312](#) (or [install-tl-unx](#))

Installation of FLTK

Note

The tar extraction directory is fltk-1.3.9 and not fltk-1.3.9-source as indicated by the tarball name.

Install FLTK by running the following commands:

```
sed -i -e '/cat./d' documentation/Makefile      &&
./configure --prefix=/usr      \
            --enable-shared  &&
make
```

If you wish to create the API documentation, issue:

```
make -C documentation html
```

The tests for the package are interactive. To execute the tests, run `test/unittests`. In addition, there are 70 other executable test programs in the `test` directory that can be run individually.

Now, as the `root` user:

```
make docdir=/usr/share/doc/fltk-1.3.9 install
```

If desired, install some example games built as a part of the tests, extra documentation and example programs. As the `root` user:

```
make -C test          docdir=/usr/share/doc/fltk-1.3.9 install-linux &&
make -C documentation docdir=/usr/share/doc/fltk-1.3.9 install-linux
```

Command Explanations

`sed ... documentation/Makefile`: Avoid installing pages in `/usr/share/man/cat*`.

Contents

Installed Programs: blocks, checkers, fltk-config, fluid, and sudoku

Installed Libraries: libfltk.{a,so}, libfltk_forms.{a,so}, libfltk_gl.{a,so}, and libfltk_images.{a,so}

Installed Directories: /usr/include/FL and /usr/share/doc/fltk-1.3.9

Short Descriptions

<code>blocks</code>	is a FLTK-based block elimination game
<code>checkers</code>	is a FLTK-based version of the game of checkers
<code>fltk-config</code>	is a utility script that can be used to get information about the current version of FLTK that is installed on the system
<code>fluid</code>	is an interactive GUI designer for FLTK
<code>sudoku</code>	is an implementation of the popular Sudoku game
<code>libfltk.so</code>	contains functions that provide an API to implement graphical user interfaces

Freeglut-3.6.0

Introduction to Freeglut

Freeglut is intended to be a 100% compatible, completely opensourced clone of the GLUT library. GLUT is a window system independent toolkit for writing OpenGL programs, implementing a simple windowing API, which makes learning about and exploring OpenGL programming very easy.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/freeglut/freeglut-3.6.0.tar.gz>
- Download MD5 sum: 1a1c4712b3100f49f5dea22a1ad57c34
- Download size: 420 KB
- Estimated disk space required: 5.2 MB
- Estimated build time: less than 0.1 SBU

Freeglut Dependencies

Required

[CMake-3.30.2](#) and [Mesa-24.1.5](#)

Recommended

[GLU-9.0.3](#)

Installation of Freeglut

Install Freeglut by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_BUILD_TYPE=Release           \
-D FREEGLUT_BUILD_DEMOS=OFF          \
-D FREEGLUT_BUILD_STATIC_LIBS=OFF    \
-W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D FREEGLUT_BUILD_DEMOS=OFF`: Disable building optional demo programs. Note that if you choose to build them, their installation must be done manually. The demo programs are limited and installation is not recommended.

`-D FREEGLUT_BUILD_STATIC_LIBS=OFF`: Do not build the static library.

Contents

Installed Programs: None

Installed Library: libglut.so

Installed Directories: /usr/lib/cmake/FreeGLUT

Short Descriptions

`libglut.so` contains functions that implement the OpenGL Utility Toolkit

gdk-pixbuf-2.42.12

Introduction to Gdk Pixbuf

The Gdk Pixbuf package is a toolkit for image loading and pixel buffer manipulation. It is used by GTK+ 3 to load and manipulate images. In the past it was distributed as part of GTK+ 2, but it was split off into a separate package in preparation for the change to GTK+ 3.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gdk-pixbuf/2.42/gdk-pixbuf-2.42.12.tar.xz>
- Download MD5 sum: f986fdbba5ec6233c96f8b6535811780
- Download size: 6.2 MB
- Estimated disk space required: 33 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Gdk Pixbuf Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection required for GNOME), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), and [shared-mime-info-2.4](#)

Recommended

[docutils-0.21.2](#), [librsvg-2.58.3](#) (runtime dependency, needed for loading symbolic icons) and [libtiff-4.6.0](#)

Optional

[Gi-DocGen-2024.1](#) (to generate documentation), [libavif-1.1.1](#) (runtime dependency, needed for loading AVIF images), [libjxl-0.10.3](#) (runtime dependency, needed for loading JPEG XL images), and [webp-pixbuf-loader-0.2.7](#) (runtime dependency, needed for loading WebP images)

Installation of Gdk Pixbuf

Install Gdk Pixbuf by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D others=enabled \
    --wrap-mode=nofallback &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/docs_dir =/s@\$@ / 'gdk-pixbuf-2.42.12'@" -i ../docs/meson.build &&
meson configure -D gtk_doc=true &&
ninja
```

To test the results, issue: `ninja test`. The tests make a heavy use of disk.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package on to your system using a “DESTDIR” method, an important file was not installed and should be copied and/or generated. Generate it using the following command as the `root` user:

```
gdk-pixbuf-query-loaders --update-cache
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`--wrap-mode=nofallback`: This switch prevents `meson` from using subproject fallbacks for any dependency declarations in the build files, stopping it downloading any optional dependency which is not installed on the system.

`-D others=enabled`: Enable the loaders for various image formats, for example BMP and XPM.

`-D man=false`: Use this option if you do not want to generate manual pages, or if you do not want to install [docutils-0.21.2](#).

Contents

Installed Programs: gdk-pixbuf-csource, gdk-pixbuf-pixdata, gdk-pixbuf-query-loaders, and gdk-pixbuf-thumbnailer

Installed Libraries: libgdk_pixbuf-2.0.so

Installed Directories: /usr/{include,lib}/gdk-pixbuf-2.0, /usr/libexec/installed-tests/gdk-pixbuf, /usr/share/installed-tests/gdk-pixbuf, /usr/share/gtk-doc/html/gdk-pixbuf, and /usr/share/thumbnailers

Short Descriptions

<code>gdk-pixbuf-csource</code>	is a small utility that generates C code containing images, used for compiling images directly into programs
<code>gdk-pixbuf-pixdata</code>	is a tool used to convert GdkPixbuf to GdkPixdata
<code>gdk-pixbuf-query-loaders</code>	collects information about loadable modules for Gdk Pixbuf and writes it to the default cache file location, or to stdout
<code>gdk-pixbuf-thumbnailer</code>	creates thumbnails of pictures for use in other applications
<code>libgdk_pixbuf-2.0.so</code>	contains functions used to load and render images

GLEW-2.2.0

Introduction to GLEW

GLEW is the OpenGL Extension Wrangler Library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/glew/glew-2.2.0.tgz>
- Download MD5 sum: 3579164bccae09e36c0af7f4fd5c7c7
- Download size: 820 KB
- Estimated disk space required: 16 MB
- Estimated build time: less than 0.1 SBU

glew Dependencies

Required

[Mesa-24.1.5](#)

Installation of GLEW

Install GLEW by running the following commands:

```
sed -i 's%lib64%lib%g' config/Makefile.linux &&
sed -i -e '/glew.lib.static:/d' \
-e '/0644 .*STATIC/d' \
-e 's/glew.lib.static//' Makefile      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install.all
```

Command Explanations

`sed -i 's%lib64%lib%g' ...`: This ensures that the library is installed in `/usr/lib`.

`sed -i -e '/glew.lib.static:/d' ...`: This suppresses the static library.

`make install.all`: This installs the programs as well as the library.

Contents

Installed Programs: glewinfo and visualinfo

Installed Library: libGLEW.so

Installed Directory: /usr/include/GL

Short Descriptions

<code>glewinfo</code>	provides information about the supported extensions
<code>visualinfo</code>	is an extended version of glxinfo
<code>libGLEW.so</code>	provides functions to access OpenGL extensions

Glslang-14.3.0

Introduction to Glslang

The Glslang package contains an frontend and validator for OpenGL, OpenGL ES, and Vulkan shaders.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KhronosGroup/glslang/archive/14.3.0/glslang-14.3.0.tar.gz>
- Download MD5 sum: 90eec8d14a8d1002389e00ab449c7ca8
- Download size: 3.7 MB
- Estimated disk space required: 174 MB (with tests)
- Estimated build time: 0.3 SBU (with parallelism=4; with tests)

Glslang Dependencies

Required

[CMake-3.30.2](#) and [SPIRV-Tools-1.3.290.0](#)

Installation of Glslang

Install Glslang by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_BUILD_TYPE=Release           \
-D ALLOW_EXTERNAL_SPIRV_TOOLS=ON    \
-D BUILD_SHARED_LIBS=ON              \
-D GLSLANG_TESTS=ON                 \
-G Ninja .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D ALLOW_EXTERNAL_SPIRV_TOOLS=ON`: This switch allows the build system to use the system-installed copy of [SPIRV-Tools-1.3.290.0](#), instead of downloading and installing it's own copy.

`-D BUILD_SHARED_LIBS=ON`: This switch builds shared versions of the libraries, and does not install static versions of them.

`-D GLSLANG_TESTS=ON`: This switch builds the tests for the package.

Contents

Installed Programs: glslang, glslang-validator (symlink to glslang), and spirv-remap

Installed Libraries: libglslang.so, libglslang-default-resource-limits.so, libSPIRV.so, and libSPVRemapper.so

Installed Directories: /usr/include/glslang and /usr/lib/cmake/glslang

Short Descriptions

<code>glslang</code>	provides a front end and validator for OpenGL, OpenGL ES, and Vulkan shaders
<code>libglslang.so</code>	contains functions that provide a front-end and validator for OpenGL, OpenGL ES, and Vulkan shaders to other programs
<code>libSPIRV.so</code>	provides a front-end and generator for SPIR-V binaries
<code>libSPVRemapper.so</code>	provides a remapper for SPIR-V binaries

GLU-9.0.3

Introduction to GLU

This package provides the Mesa OpenGL Utility library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.mesa3d.org/glu/glu-9.0.3.tar.xz>
- Download MD5 sum: 06a4fff9179a98ea32ef41b6d83f6b19
- Download size: 216 KB
- Estimated disk space required: 5.9 MB
- Estimated build time: 0.2 SBU

GLU Dependencies

Required

[Mesa-24.1.5](#)

Installation of GLU

Install GLU by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=$XORG_PREFIX \
    -D gl_provider=gl \
    --buildtype=release &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install &&
rm -vf /usr/lib/libGLU.a
```

Contents

Installed Programs: None

Installed Library: libGLU.so

Installed Directories: None

Short Descriptions

`libGLU.so` is the Mesa OpenGL Utility library

GOffice-0.10.57

Introduction to GOffice

The GOffice package contains a library of GLib/GTK document centric objects and utilities. This is useful for performing common operations for document centric applications that are conceptually simple, but complex to implement fully. Some of the operations provided by the GOffice library include support for plugins, load/save routines for application documents and undo/redo functions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/goffice/0.10/goffice-0.10.57.tar.xz>
- Download MD5 sum: 4c55f99d8fafef91276cb7538521fdb
- Download size: 2.4 MB
- Estimated disk space required: 81 MB (with tests)

- Estimated build time: 0.4 SBU (Using parallelism=4; with tests)

GOffice Dependencies

Required

[GTK+-3.24.43](#), [libgsf-1.14.52](#), [librsvg-2.58.3](#), [libsslt-1.1.42](#), and [Which-2.21](#)

Optional

[GLib-2.80.4](#) (with GObject Introspection), [ghostscript-10.03.1](#), [gsettings-desktop-schemas-46.1](#), [GTK-Doc-1.34.0](#), [Lasem](#), and [libspectre](#)

Installation of GOffice

Install GOffice by running the following commands:

```
./configure --prefix=/usr &&
make
```

If you wish to run the tests, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Libraries: libgoffice-0.10.so

Installed Directories: /usr/include/libgoffice-0.10, /usr/{lib,share}/goffice, and /usr/share/gtk-doc/html/goffice-0.10

Short Descriptions

`libgoffice-0.10.so` contains API functions to provide support for document centric objects and utilities

Grantlee-5.3.1

Introduction to grantlee

Grantlee is a set of free software libraries written using the Qt framework. Currently two libraries are shipped with Grantlee: Grantlee Templates and Grantlee TextDocument. The goal of Grantlee Templates is to make it easier for application developers to separate the structure of documents from the data they contain, opening the door for theming.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/steveire/grantlee/releases/download/v5.3.1/grantlee-5.3.1.tar.gz>
- Download MD5 sum: 4ef8eae5dd61e3c7603d76208eb4d922
- Download size: 1.1 MB
- Estimated disk space required: 27 MB
- Estimated build time: 0.4 SBU (Using parallelism=4)

Grantlee Dependencies

Required

[CMake-3.30.2](#) and [qt5-components-5.15.14](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of Grantlee

Install Grantlee by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
.. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: none

Installed Libraries: libGrantlee_Templates.so and libGrantlee_TextDocument.so

Installed Directories: /usr/lib/cmake/Grantlee5, /usr/lib/grantlee/5.2, and /usr/include/grantlee

Short Descriptions

libGrantlee_Templates.so

contains common documentation templates for separating documents from their structure

libGrantlee_TextDocument.so

contains functions that allow separating the content of text documents from their structure.

Graphene-1.10.8

Introduction to Graphene

The Graphene package provides a thin layer of types for graphics libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/graphene/1.10/graphene-1.10.8.tar.xz>
- Download MD5 sum: 169e3c507b5a5c26e9af492412070b81
- Download size: 328 KB
- Estimated disk space required: 7.6 MB
- Estimated build time: less than 0.1 SBU (with tests)

Graphene Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection)

Installation of Graphene

Install Graphene by running the following commands:

```
mkdir build &&
cd      build &&
```

```
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this switch if you have [GTK-Doc-1.34.0](#) installed and wish to generate the API documentation.

Contents

Installed Programs: None

Installed Libraries: libgraphene-1.0.so

Installed Directories: /usr/include/graphene-1.0, /usr/lib/graphene-1.0, /usr/{libexec,share}/installed-tests/graphene-1.0

Short Descriptions

`libgraphene-1.0.so` contains functions that provide a thin layer of types for graphics libraries

GTK+-3.24.43

Introduction to GTK+ 3

The GTK+ 3 package contains libraries used for creating graphical user interfaces for applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtk+/3.24/gtk+-3.24.43.tar.xz>
- Download MD5 sum: 226e372664a7fae749a50a04983ecd8d
- Download size: 13 MB
- Estimated disk space required: 252 MB (add 8 MB for tests)
- Estimated build time: 0.8 SBU (using parallelism=4; add 0.2 SBU for tests)

GTK+ 3 Dependencies

Required

[at-spi2-core-2.52.0](#), [gdk-pixbuf-2.42.12](#), [libepoxy-1.5.10](#), and [Pango-1.54.0](#)

Recommended

[adwaita-icon-theme-46.2](#) (at runtime; default for some gtk+3 settings keys and also needed for one test), [docbook-xsl-nons-1.79.2](#) (for generating manual pages), [hicolor-icon-theme-0.18](#) (needed for tests), [ISO Codes-4.16.0](#), [libxkbcommon-1.7.0](#), [libxslt-1.1.42](#) (for generating manual pages), [sassc-3.6.2](#), [Wayland-1.23.0](#), and [wayland-protocols-1.36](#)

Recommended (Required if building GNOME)

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[colord-1.4.7](#), [Cups-2.4.10](#), [GTK-Doc-1.34.0](#), [libcloudproviders-0.3.6](#), [PyAtSpi2-2.46.1](#) (for tests), [Tracker-3.7.3](#), and [PAPI](#)

Installation of GTK+ 3

Install GTK+ 3 by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D man=true \
    -D broadway_backend=true &&
ninja
```

To test the results you need a graphical session, then issue `dbus-run-session ninja test`.

Now, as the `root` user:

```
ninja install
```

One test named `check-cursor-names` is known to fail if [adwaita-icon-theme-46.2](#) is not installed.

Note

If you installed the package on to your system using a “DESTDIR” method, an important file was not installed and must be copied and/or generated. Generate it using the following command as the `root` user:

```
gtk-query-immodules-3.0 --update-cache
```

Note

If you installed the package to your system using a “DESTDIR” method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Command Explanations

`-D broadway_backend=true`: This switch enables the HTML5 GTK backend.

`-D man=true`: This switch allows generating manual pages.

`-D gtk_doc=true`: This switch enables building documentation. It requires [GTK-Doc-1.34.0](#).

`-D tracker3=true`: This switch enables the search function based on Tracker 3 in the GTK+-3 file chooser dialog. It requires [Tracker-3.7.3](#).

`-D cloudproviders=true`: Use this switch if you have [libcloudproviders-0.3.6](#) installed and wish to enable support for cloud providers in a file chooser window.

Configuring GTK+ 3

Config Files

`~/.config/gtk-3.0/settings.ini` and `/etc/gtk-3.0/settings.ini`

Configuration Information

GTK+ 3 themes change the way a GTK+ 3 application looks. An icon theme can be used to change the icons that appear on the application's toolbar. If you have installed a GTK+ 3 theme (e.g. the Adwaita theme built in GTK+ 3), an icon theme (such as [oxygene-icons-6.0.0](#)) and/or a font ([Dejavu fonts](#)), you can set your preferences in `~/.config/gtk-3.0/settings.ini`, or the default system wide configuration file (as the `root` user), in `/etc/gtk-3.0/settings.ini`. For the local user an example is:

```
mkdir -vp ~/.config/gtk-3.0
cat > ~/.config/gtk-3.0/settings.ini << "EOF"
[Settings]
```

```

gtk-theme-name = Adwaiata
gtk-icon-theme-name = oxygen
gtk-font-name = DejaVu Sans 12
gtk-cursor-theme-size = 18
gtk-toolbar-style = GTK_TOOLBAR_BOTH_HORIZ
gtk-xft-antialias = 1
gtk-xft-hinting = 1
gtk-xft-hintstyle = hintslight
gtk-xft-rgba = rgb
gtk-cursor-theme-name = Adwaiata
EOF

```

There are many settings keys, some with default values. You can find them at [Settings: GTK+ 3 Reference Manual](#). There are many more themes available at <https://www.gnome-look.org/browse/> and other places.

As part of GTK+-3.0's redesign, the scroll bar buttons are no longer visible on the scrollbar in many applications. If this functionality is desired, modify the `gtk.css` file and restore them using the following command:

```

cat > ~/.config/gtk-3.0/gtk.css << "EOF"
* {
  -GtkScrollbar-has-backward-stepper: 1;
  -GtkScrollbar-has-forward-stepper: 1;
}
EOF

```

Contents

- Installed Programs:** broadwayd, gtk3-demo, gtk3-demo-application, gtk3-icon-browser, gtk3-widget-factory, gtk-builder-tool, gtk-encode-symbolic-svg, gtk-launch, gtk-query-immodules-3.0, gtk-query-settings, and gtk-update-icon-cache
- Installed Libraries:** libgailutil-3.so, libgdk-3.so, and libgtk-3.so
- Installed Directories:** /etc/gtk-3.0, /usr/include/{gail,gtk}-3.0, /usr/{lib,share}/gtk-3.0, and /usr/share/themes/{Default,Emacs}/gtk-3.0

Short Descriptions

<code>broadwayd</code>	provides support for displaying GTK+ 3 applications in a web browser, using HTML5 and web sockets
<code>gtk3-demo</code>	is a simple program that demonstrates some of the tasks that can be done with GTK+ 3
<code>gtk3-demo-application</code>	is a simple GTK+ 3 application
<code>gtk3-icon-browser</code>	is a utility to explore the icons in the current icon theme. It shows icons in various sizes, their symbolic variants where available, as well as a description of the icon and its context
<code>gtk3-widget-factory</code>	is a program to view GTK+ 3 themes and widgets
<code>gtk-builder-tool</code>	performs various operations on GtkBuilder .ui files
<code>gtk-encode-symbolic-svg</code>	converts symbolic SVG icons into specially prepared PNG files. GTK+ 3 can load and recolor these PNGs, just like original SVGs, but loading them is much faster
<code>gtk-launch</code>	launches an application using the given name. The name should match the application desktop file name, as residing in <code>/usr/share/applications</code> , with or without the '.desktop' suffix
<code>gtk-query-immodules-3.0</code>	collects information about loadable input method modules for GTK+ 3 and writes it to the default cache file location, or to standard output
<code>gtk-query-settings</code>	provides a complete listing of all settings related to GTK+ 3
<code>gtk-update-icon-cache</code>	is an icon theme caching utility that creates mmap()able cache files for icon themes
<code>libgailutil-3.so</code>	contains functions that implement the accessibility interfaces defined by the GNOME Accessibility Toolkit
<code>libgdk-3.so</code>	contains functions that act as a wrapper around the low-level drawing and windowing functions provided by the underlying graphics system
<code>libgtk-3.so</code>	contains functions that provide an API to implement graphical user interfaces

GTK-4.14.5

Introduction to GTK 4

The GTK 4 package contains libraries used for creating graphical user interfaces for applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtk/4.14/gtk-4.14.5.tar.xz>
- Download MD5 sum: 4d7c8edb0aed4048053ca9f805ade4
- Download size: 17 MB
- Estimated disk space required: 711 MB (with docs; add 66 MB for tests)
- Estimated build time: 1.3 SBU (using parallelism=4; with docs; add 1.3 SBU for tests)

GTK 4 Dependencies

Required

[FriBidi-1.0.15](#), [gdk-pixbuf-2.42.12](#), [graphene-1.10.8](#), [ISO Codes-4.16.0](#), [libePOXY-1.5.10](#), [libXkbcommon-1.7.0](#), [Pango-1.54.0](#), [PyGObject-3.48.2](#), and [wayland-protocols-1.36](#)

Recommended

[adwaita-icon-theme-46.2](#) (default for some gtk4 settings keys), [gst-plugins-bad-1.24.7](#), [gst-plugins-good-1.24.7](#) (runtime, built with [libvpx-1.14.1](#)), [hicolor-icon-theme-0.18](#) (needed for tests and for defaults), and [librsvg-2.58.3](#)

Recommended (Required if building GNOME)

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[colord-1.4.7](#), [Cups-2.4.10](#), [docutils-0.21.2](#), [Gi-DocGen-2024.1](#), [Highlight-4.13](#) (runtime, only used by [gtk4-demo](#) for syntax highlighting of demo source code), [libcloudproviders-0.3.6](#), [sassc-3.6.2](#), [Tracker-3.7.3](#), [Vulkan-Loader-1.3.294](#), [cpdb](#), [glslc](#), and [sysprof](#)

Installation of GTK 4

Install GTK 4 by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D broadway-backend=true \
            -D introspection=enabled \
            -D vulkan=disabled \
            .. &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "s@'doc'@& / 'gtk-4.14.5'@" -i ..../docs/reference/meson.build &&
meson configure -D documentation=true &&
ninja
```

To run the tests, issue: `dbus-run-session meson test --setup x11`. If you are in a Wayland session, replace x11 with wayland. Nine tests are known to fail if [Cantrell fonts](#) are not installed. Many tests will fail if `~/.config/gtk-4.0/settings.ini` exists and the gtk-modules line is not commented out. On systems with NVIDIA graphics cards, the tests may take significantly longer than the above test time.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D broadway-backend=true`: This switch enables the HTML5 GDK backend.

`-D introspection=enabled`: This switch enables using GObject Introspection to generate GIR bindings of this packages. These bindings are required by GNOME.

`-D cloudproviders=enabled`: Use this switch if you have [libcloudproviders-0.3.6](#) installed and wish to enable support for cloud providers in a file chooser window.

`-D tracker=enabled`: Use this switch if you have [Tracker-3.7.3](#) installed and wish to use search functionality when running a file chooser.

`-D colord=enabled`: Use this switch if you have [colord-1.4.7](#) installed and wish to use colord with the CUPS printing backend.

`-D man-pages=true`: Use this switch if you have [docutils-0.21.2](#) installed and wish to generate the man pages.

`-D sysprof=enabled`: Use this switch if you have [sysprof](#) installed and wish to enable tracing support for GTK4-based applications.

`-D vulkan=enabled`: Use this switch if you have [Vulkan-Loader-1.3.294](#) installed and wish to build the experimental Vulkan backend. Note that you must have [glslc](#) installed if you decide to enable this option.

Configuring GTK 4

Config Files

`~/.config/gtk-4.0/settings.ini` and `/usr/share/gtk-4.0/settings.ini`

Configuration Information

GTK 4 themes change the way a GTK 4 application looks. An icon theme can be used to change the icons that appear on the application's toolbar. If you have installed a GTK 4 theme (e.g., the Adwaita theme built in GTK 4), an icon theme (such as [oxygen-icons-6.0.0](#)) and/or a font ([DejaVu fonts](#)), you can set your preferences in `~/.config/gtk-4.0/settings.ini`, or the default system-wide configuration file (as the `root` user), in `/usr/share/gtk-4.0/settings.ini`. For the local user, an example is:

```
mkdir -pv ~/.config/gtk-4.0
cat > ~/.config/gtk-4.0/settings.ini << "EOF"
[Settings]
gtk-theme-name = Adwaita
gtk-icon-theme-name = oxygen
gtk-font-name = DejaVu Sans 12
gtk-cursor-theme-size = 18
gtk-xft-antialias = 1
gtk-xft-hinting = 1
gtk-xft-hintstyle = hintslight
gtk-xft-rgba = rgb
gtk-cursor-theme-name = Adwaita
EOF
```

There are many settings keys, some with default values. You can find them at [Settings: GTK 4 Reference Manual](#).

Contents

Installed Programs: `gtk4-broadwayd`, `gtk4-builder-tool`, `gtk4-demo`, `gtk4-demo-application`, `gtk4-encode-symbolic-svg`, `gtk4-icon-browser`, `gtk4-launch`, `gtk4-node-editor`, `gtk4-print-editor`, `gtk4-query-settings`, `gtk4-update-icon-cache`, and `gtk4-widget-factory`

Installed Libraries: `libgtk-4.so`

Installed Directories: `/usr/include/gtk-4.0`, `/usr/lib/gtk-4.0`, and `/usr/share/gtk-4.0`

Short Descriptions

<code>gtk4-broadwayd</code>	provides support for displaying GTK 4 applications in a web browser using HTML5 and web sockets
<code>gtk4-builder-tool</code>	performs various operations on GtkBuilder .ui files
<code>gtk4-demo</code>	is a simple program that demonstrates some of the tasks that can be done with GTK 4
<code>gtk4-demo-application</code>	is a simple GTK 4 application which is useful for testing
<code>gtk4-encode-symbolic-svg</code>	converts symbolic SVG icons into special PNG files. GTK 4 can load and recolor these PNGs, just like original SVGs, but loading them is much faster

<code>gtk4-icon-browser</code>	is a utility to explore the icons in the current icon theme. It shows icons in various sizes, their symbolic variants when available, as well as a description of the icon and its context
<code>gtk4-launch</code>	launches an application using the given name. The name should match the application .desktop file name (as seen in <code>/usr/share/applications</code>), with or without the '.desktop' extension
<code>gtk4-node-editor</code>	is a utility to show and edit render node files. Such render node files can be obtained e.g. from the GTK inspector
<code>gtk4-print-editor</code>	is a simple program to demonstrate printing using GTK 4 applications
<code>gtk4-query-settings</code>	provides a complete listing of all settings related to GTK 4
<code>gtk4-update-icon-cache</code>	is an icon theme caching utility that creates mmap()able cache files for icon themes
<code>gtk4-widget-factory</code>	is a program to view GTK 4 themes and widgets
<code>libgtk-4.so</code>	contains functions that provide an API to implement graphical user interfaces

Gtkmm-3.24.9

Introduction to Gtkmm

The Gtkmm package provides a C++ interface to GTK+ 3.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtkmm/3.24/gtkmm-3.24.9.tar.xz>
- Download MD5 sum: 47871a7973e186c1189b2145b507de15
- Download size: 14 MB
- Estimated disk space required: 200 MB (with tests)
- Estimated build time: 1.2 SBU (Using parallelism=4; with tests)

Gtkmm Dependencies

Required

[Atkmm-2.28.4](#), [GTK+-3.24.43](#), and [Pangomm-2.46.4](#)

Optional

[Doxygen-1.12.0](#)

Installation of Gtkmm

Install Gtkmm by running the following commands:

```
mkdir gtkmm3-build &&
cd gtkmm3-build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`. Note that you must be in a graphical environment, as the tests try to open some windows.

Now, as the `root` user:

```
ninja install
```

If you have built the documentation (see Command Explanations below) it was installed to `/usr/share/doc/gtkmm-3.0`. For consistency, move it to a versioned directory as the `root` user:

```
mv -v /usr/share/doc/gtkmm-3.0 /usr/share/doc/gtkmm-3.24.9
```

Command Explanations

-D build-documentation=true: If you have installed [Doxygen-1.12.0](#) this define will build and install the documentation.

Contents

Installed Programs: None

Installed Libraries: libgdkmm-3.0.so and libgtkmm-3.0.so

Installed Directories: /usr/include/gdkmm-3.0, /usr/include/gtkmm-3.0, /usr/lib/gdkmm-3.0, /usr/lib/gtkmm-3.0, and optionally /usr/share/{devhelp/books/gtkmm-3.0,doc/gtkmm-3.24.9}

Short Descriptions

libgdkmm-3.0.so	contains the GDK API classes
libgtkmm-3.0.so	contains the GTK+ 3 API classes

Gtkmm-4.14.0

Introduction to Gtkmm

The Gtkmm package provides a C++ interface to GTK-4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtkmm/4.14/gtkmm-4.14.0.tar.xz>
- Download MD5 sum: 2e25e9c3f64793ff8b51bdd24196d78f
- Download size: 16 MB
- Estimated disk space required: 206 MB (with tests)
- Estimated build time: 1.3 SBU (Using parallelism=4; with tests)

Gtkmm Dependencies

Required

[Atkmm-2.28.4](#), [GTK-4.14.5](#), and [Pangomm-2.54.0](#)

Optional

[Doxygen-1.12.0](#) and [Vulkan-Loader-1.3.294](#)

Installation of Gtkmm

Install Gtkmm by running the following commands:

```
mkdir gtkmm4-build &&
cd gtkmm4-build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`. Note that you must be in a graphical environment, as the tests try to open some windows.

Now, as the `root` user:

```
ninja install
```

If you have built the documentation (see Command Explanations below) it was installed to `/usr/share/doc/gtkmm-4.0`. For consistency, move it to a versioned directory as the `root` user:

```
mv -v /usr/share/doc/gtkmm-4.0 /usr/share/doc/gtkmm-4.14.0
```

Command Explanations

-D build-documentation=true: If you have installed [Doxygen-1.12.0](#) this define will build and install the documentation.

Contents

Installed Programs: None

Installed Libraries: libgdkmm-4.0.so and libgtkmm-4.0.so

Installed Directories: /usr/include/gdkmm-4.0, /usr/include/gtkmm-4.0, /usr/lib/gdkmm-4.0, /usr/lib/gtkmm-4.0, and optionally /usr/share/{devhelp/books/gtkmm-4.0,doc/gtkmm-4.14.0}

Short Descriptions

libgdkmm-4.0.so	contains the GDK API classes
libgtkmm-4.0.so	contains the GTK 4 API classes

gtk-vnc-1.3.1

Introduction to Gtk VNC

The Gtk VNC package contains a VNC viewer widget for GTK+. It is built using coroutines allowing it to be completely asynchronous while remaining single threaded.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtk-vnc/1.3/gtk-vnc-1.3.1.tar.xz>
- Download MD5 sum: 27f0dc7f33cbfdcaa6c9eb7cf169f4866
- Download size: 220 KB
- Estimated disk space required: 5.2 MB
- Estimated build time: 0.1 SBU

Gtk VNC Dependencies

Required

[GnuTLS-3.8.7.1](#), [GTK+-3.24.43](#), and [libgcrypt-1.11.0](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[Cyrus SASL-2.1.28](#) and [PulseAudio-17.0](#)

Installation of Gtk VNC

Install Gtk VNC by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D with-vala=false`: This switch disables building of the Vala bindings. Add this if you decide to build gtk-vnc without vala installed.

Contents

Installed Program: gvnccapture

Installed Libraries: libgtk-vnc-2.0.so, libgvnc-1.0.so and libgvncpulse-1.0.so

Installed Directories: /usr/include/gtk-vnc-2.0, /usr/include/gvnc-1.0 and /usr/include/gvncpulse-1.0

Short Descriptions

<code>gvnccapture</code>	is used to capture image from VNC server
<code>libgtk-vnc-2.0.so</code>	contains the GTK+ 3 bindings for Gtk VNC
<code>libgvnc-1.0.so</code>	contains the GObject bindings for Gtk VNC
<code>libgvncpulse-1.0.so</code>	is the PulseAudio bridge for Gtk VNC

gtksourceview-3.24.11

Introduction to GtkSourceView

The GtkSourceView package contains libraries used for extending the GTK+ text functions to include syntax highlighting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtksourceview/3.24/gtksourceview-3.24.11.tar.xz>
- Download MD5 sum: b748da426a7d64e1304f0c532b0f2a67
- Download size: 1.4 MB
- Estimated disk space required: 80 MB (with tests)
- Estimated build time: 0.2 SBU (using parallelism=4; with tests)

GtkSourceView Dependencies

Required

[GTK+-3.24.43](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[Vala-0.56.17](#), [Valgrind-3.23.0](#), [GTK-Doc-1.34.0](#), [itstool-2.0.7](#), [fop-2.9](#) (or [dblatex](#)), and [Glade](#)

Installation of GtkSourceView

First, fix building this package with gcc-14:

```
sed -i 's/g_object_ref (buffer)/g_object_ref (GTK_SOURCE_BUFFER (buffer))/' gtksourceview/gtksourceview.c
```

Install GtkSourceView by running the following commands:

```
./configure --prefix=/usr &&  
make
```

To test the results, issue `make check`. The tests need to be run in a graphical environment.

Now, as the `root` user:

```
make install
```

Command Explanations

--enable-gtk-doc: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Library: libgtksourceview-3.0.so

Installed Directories: /usr/{include,share,share/gtk-doc/html}/gtksourceview-3.0

Short Descriptions

libgtksourceview-3.0.so contains function extensions for the GtkTextView widget

gtksourceview4-4.8.4

Introduction to GtkSourceView

The GtkSourceView package contains libraries used for extending the GTK+ text functions to include syntax highlighting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtksourceview/4.8/gtksourceview-4.8.4.tar.xz>
- Download MD5 sum: 2bf056caaae27654ec3a5930dd5597d3
- Download size: 1.1 MB
- Estimated disk space required: 71 MB (with tests)
- Estimated build time: 0.1 SBU (with tests; both using parallelism=4)

GtkSourceView Dependencies

Required

[GTK+-3.24.43](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [libxml2-2.13.3](#)

Optional

[Vala-0.56.17](#), [Valgrind-3.23.0](#), [GTK-Doc-1.34.0](#), [itstool-2.0.7](#), [fop-2.9](#) (or [dblatex](#)), and [Glade](#)

Installation of GtkSourceView

Install GtkSourceView by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue `ninja test`. The tests need to be run in a graphical environment.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D gtk_doc=true`: This option is normally used if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Library: libgtksourceview-4.so

Installed Directories: /usr/{include,share,share/gtk-doc/html}/gtksourceview-4

Short Descriptions

libgtksourceview-4.so contains function extensions for the GtkTextView widget

gtksourceview5-5.12.1

Introduction to GtkSourceView

The GtkSourceView package contains a library used for extending the GTK text functions to include syntax highlighting.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gtksourceview/5.12/gtksourceview-5.12.1.tar.xz>
- Download MD5 sum: 1a2e5289b6f37032271b3f96d8034751
- Download size: 1.2 MB
- Estimated disk space required: 47 MB
- Estimated build time: 0.2 SBU (with tests; both using parallelism=4)

GtkSourceView Dependencies

Required

[GTK-4.14.5](#) and [pcre2-10.44](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [libxml2-2.13.3](#)

Optional

[Gi-DocGen-2024.1](#), [Vala-0.56.17](#), [Valgrind-3.23.0](#), [Vulkan-Loader-1.3.294](#), and [sysprof](#)

Installation of GtkSourceView

Install GtkSourceView by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D documentation=true`: Use this switch if you have [Gi-DocGen-2024.1](#) installed and wish to generate the API documentation.

`-D sysprof=true`: Use this switch if you have [sysprof](#) installed and wish to build with sysprof profiler support.

Contents

Installed Programs: None

Installed Libraries: libgtksourceview-5.so

Installed Directories: /usr/include/gtksourceview-5 and /usr/share/gtksourceview-5

Short Descriptions

libgtksourceview-5.so contains function extensions for the GtkTextView widget

imlib2-1.12.3

Introduction to imlib2

imlib2 is a graphics library for fast file loading, saving, rendering and manipulation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/enlightenment/imlib2-1.12.3.tar.xz>
- Download MD5 sum: 93e5b769ed02a183dfd78569f7b0fbe3
- Download size: 816 KB
- Estimated disk space required: 13 MB (with docs)
- Estimated build time: 0.2 SBU (with docs)

imlib2 Dependencies

Required

[Xorg Libraries](#)

Recommended

[giflib-5.2.2](#) and [librsvg-2.58.3](#)

Optional

[Doxygen-1.12.0](#) (for API documentation), [highway-1.2.0](#), [libjpeg-turbo-3.0.1](#), [libjxl-0.10.3](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [x265-3.6](#), [libheif](#), [libid3tag](#), [libspectre](#)

Installation of imlib2

Install imlib2 by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you generated the API documentation, install it as the `root` user:

```
install -v -m755 -d          /usr/share/doc/imlib2-1.12.3/html &&  
install -v -m644   doc/html/* /usr/share/doc/imlib2-1.12.3/html
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: imlib2_bumpmap, imlib2_colorspace, imlib2_conv, imlib2_grab, imlib2_load, imlib2_poly, imlib2_show, imlib2_test, and imlib2_view

Installed Libraries: libImlib2.so and various filters and image loader modules.

Installed Directories: /usr/lib/imlib2, /usr/share/doc/imlib2-1.12.3, and /usr/share/imlib2

Short Descriptions

imlib2_bumpmap	is to test the imlib2 bumpmap function
imlib2_colorspace	is to test the imlib2 colorspace function
imlib2_conv	converts images between formats
imlib2_grab	takes screenshots
imlib2_load	loads and caches images
imlib2_poly	is to test the imlib2 poly function
imlib2_show	is an imlib2 program test
imlib2_test	is an imlib2 program test
imlib2_view	displays image files
libImlib2.so	provides the functions for programs to handle various image data formats

kColorPicker-0.3.1

Introduction to kColorPicker

kColorPicker is a QToolButton library with a color popup menu, which lets you select colors. The popup menu features a color dialog button which can be used to add custom colors to the popup menu.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/ksnip/kColorPicker/archive/v0.3.1/kColorPicker-0.3.1.tar.gz>
- Download MD5 sum: 1efc91252446af0d7e5c467ea7d517e7
- Download size: 16 KB
- Estimated disk space required: 2.4 MB
- Estimated build time: less than 0.1 SBU

kColorPicker Dependencies

Required

[CMake-3.30.2](#) and [Qt-6.7.2](#)

Installation of kColorPicker

Install kColorPicker by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_SHARED_LIBS=ON \
-D BUILD_WITH_QT6=ON \
.. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libkColorPicker.so

Installed Directories: /usr/lib/cmake/kColorPicker

kImageAnnotator-0.7.1

Introduction to kImageAnnotator

kImageAnnotator is a tool for annotating images.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/ksnip/kImageAnnotator/archive/v0.7.1/kImageAnnotator-0.7.1.tar.gz>
- Download MD5 sum: 68990dfe7fe03f1aff5e0e5338b9f3bb
- Download size: 264 KB
- Estimated disk space required: 31 MB
- Estimated build time: 0.6 SBU (Using parallelism=4)

kImageAnnotator Dependencies

Required

[kcolorpicker-0.3.1](#)

Installation of kImageAnnotator

Install kImageAnnotator by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_SHARED_LIBS=ON \
      -D BUILD_WITH_QT6=ON \
      ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libkImageAnnotator.so

Installed Directories: /usr/include/kImageAnnotator-Qt6, /usr/lib/cmake/kImageAnnotator-Qt6, and
/usr/share/kImageAnnotator

keybinder-3.0-0.3.2

Introduction to keybinder-3.0

The keybinder-3.0 package contains a utility library registering global X keyboard shortcuts for GTK+-3.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/kupferlauncher/keybinder/releases/download/keybinder-3.0-v0.3.2/keybinder-3.0-0.3.2.tar.gz>
- Download MD5 sum: 97260321fda721fce799174ea6ba10cf
- Download size: 370 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: less than 0.1 SBU

keybinder-3.0 Dependencies

Required

[GTK+-3.24.43](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#)

Installation of keybinder-3.0

Install keybinder-3.0 by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Library: libkeybinder-3.0.so

Installed Directories: /usr/include/keybinder-3.0 and /usr/share/gtk-doc/html/keybinder-3.0

Short Descriptions

libkeybinder-3.0.so is the library that registers global X keyboard shortcuts

libadwaita-1.5.3

Introduction to libadwaita

The libadwaita package provides additional GTK4 UI widgets for use in developing user interfaces. It is used primarily for GNOME applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libadwaita/1.5/libadwaita-1.5.3.tar.xz>

- Download MD5 sum: 9be9ebe76f086a8535ff30918926dd07
- Download size: 4.3 MB
- Estimated disk space required: 73 MB (with tests)
- Estimated build time: 0.4 SBU (with parallelism=4; with tests)

libadwaita Dependencies

Required

[AppStream-1.0.3](#) and [GTK-4.14.5](#)

Recommended

[Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#)

Installation of libadwaita

Install libadwaita by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "s/apiversion/'1.5.3'/" -i ./doc/meson.build &&
meson configure -D gtk_doc=true &&
ninja
```

To test the results, issue: `ninja test`. The tests must be run from a graphical session.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: adwaita-1-demo

Installed Libraries: libadwaita-1.so

Installed Directories: /usr/include/libadwaita-1

Short Descriptions

<code>adwaita-1-demo</code>	provides an example of how to use the libadwaita library
<code>libadwaita-1.so</code>	provides additional GTK widgets for use in creating user interfaces

libei-1.3.0

Introduction to libei

The libei package contains a set of libraries for handling emulated inputs. It is primarily aimed at the Wayland stack.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/libinput/libei/-/archive/1.3.0/libei-1.3.0.tar.gz>
- Download MD5 sum: aeaffcb5afb5ad0bb9981eb93c4cd610
- Download size: 228 KB
- Estimated disk space required: 5.4 MB
- Estimated build time: less than 0.1 SBU

libei Dependencies

Required

[attrs-24.2.0](#)

Optional

[libevdev-1.13.2](#), [libxkbcommon-1.7.0](#), [libxml2-2.13.3](#), [munit](#), and [structlog](#)

Installation of libei

Install libei by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D tests=disabled &&
ninja
```

This package does come with a test suite, but it requires an external dependency. If you have both [munit](#) and [structlog](#) installed, and you wish to run the test suite, run the following commands:

```
meson configure -D tests=enabled .. &&
ninja test
```

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D tests=disabled`: This switch avoids a dependency on [munit](#). Remove this switch if you have munit installed and wish to run the test suite.

Contents

Installed Programs: None

Installed Libraries: libei.so, libeis.so, and liboeffis.so

Installed Directories: /usr/include/libei-1.0

Short Descriptions

libei.so	provides a client side implementation for handling Emulated Input
libeis.so	provides a server side implementation for handling Emulated Input
liboeffis	provides DBus communication services between libei and the XDG RemoteDesktop portal

libgedit-amtk-5.8.0

Introduction to libgedit-amtk

The libgedit-amtk package contains a basic GTKUIManager replacement based on GAction.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gedit-technology.net/tarballs/libgedit-amtk/libgedit-amtk-5.8.0.tar.xz>
- Download MD5 sum: 1b2ad96c189dd1749b6526c6026e0618
- Download size: 60 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: 0.1 SBU

libgedit-amtk Dependencies

Required

[GTK+-3.24.43](#)

Optional

[GTK-Doc-1.34.0](#) (for documentation) and [Valgrind-3.23.0](#)

Installation of libgedit-amtk

Install libgedit-amtk by running the following commands:

```
mkdir amtk-build &&
cd amtk-build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gtk_doc=false &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: This switch prevents the build system from generating and installing the API documentation. Omit this switch if you have [GTK-Doc-1.34.0](#) installed and wish to generate and install the API documentation.

Contents

Installed Programs: None

Installed Libraries: libgedit-amtk-5.so

Installed Directories: /usr/include/libgedit-amtk-5 and /usr/share/gtk-doc/html/libgedit-amtk-5.0

Short Descriptions

libgedit-amtk-5.so provides a basic GTKUIManager replacement API based on GAction

libgedit-gtksourceview-299.2.1

Introduction to libgedit-gtksourceview

The libgedit-gtksourceview package provides a library that extends GtkTextView. This library adds support for syntax highlighting, undo/redo, file loading and saving, search and replace, a completion system, printing, displaying line numbers, and other features typical of a source code editor.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/gedit-technology/libgedit-gtksourceview/releases/download/299.2.1/libgedit-gtksourceview-299.2.1.tar.xz>
- Download MD5 sum: d92e9d2b10478569e0b02fa347b6ef5f
- Download size: 636 KB
- Estimated disk space required: 22 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

libgedit-gtksourceview Dependencies

Required

[GTK+-3.24.43](#) and [libxml2-2.13.3](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of libgedit-gtksourceview

Install libgedit-gtksourceview by running the following commands:

```
mkdir build-libgedit &&
cd build-libgedit &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false   \
            ... &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: Disables building the API documentation. Remove if you have [GTK-Doc-1.34.0](#) installed and wish to build the API documentation.

Contents

Installed Programs: None

Installed Libraries: libgedit-gtksourceview-300.so

Installed Directories: /usr/include/libgedit-gtksourceview-300 and /usr/share/libgedit-gtksourceview-300

Short Descriptions

`libgedit-gtksourceview-300.so` contains convenience functions for text editing.

libhandy-1.8.3

Introduction to libhandy

The libhandy package provides additional GTK UI widgets for use in developing user interfaces.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libhandy/1.8/libhandy-1.8.3.tar.xz>
- Download MD5 sum: af586a91ff6d4093a6e7e283dfab5f7f
- Download size: 1.8 MB
- Estimated disk space required: 24 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

libhandy Dependencies

Required

[GTK+-3.24.43](#)

Recommended

[Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#) and [Glade](#)

Installation of libhandy

Install libhandy by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`. The tests must be run from a graphical session.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this option if you have [GTK-Doc-1.34.0](#) installed and wish to install the documentation.

Contents

Installed Programs: handy-1-demo

Installed Libraries: libhandy-1.so

Installed Directories: /usr/include/libhandy-1

Short Descriptions

handy-1-demo	provides an example of how to use the libhandy library
libhandy-1.so	provides additional GTK widgets for use in creating user interfaces

Libdrm-2.4.122

Introduction to Libdrm

Libdrm provides a userspace library for accessing the direct rendering manager (DRM) on operating systems that support the ioctl interface. Libdrm is a low-level library, typically used by graphics drivers such as the Mesa DRI drivers, the X drivers, libva and similar projects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://dri.freedesktop.org/libdrm/libdrm-2.4.122.tar.xz>
- Download MD5 sum: 143c8df50e09cd1eeb1fb53f05ecb64a
- Download size: 472 KB
- Estimated disk space required: 8.7 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

libdrm Dependencies

Recommended

[Xorg Libraries](#) (for Intel KMS API support required by Mesa)

Optional

[Cairo-1.18.0](#) (for tests), [CMake-3.30.2](#) (could be used to find dependencies without pkgconfig files), [docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [docutils-0.21.2](#), and [libxslt-1.1.42](#) (to build manual pages), [libatomic_ops-7.8.2](#) (required by architectures without native atomic operations), [Valgrind-3.23.0](#), and [CUnit](#) (for AMDGPU tests)

Installation of Libdrm

Install libdrm by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=$XORG_PREFIX \
    --buildtype=release \
    -D udev=true \
    -D valgrind=disabled \
    ..
    &&
ninja
```

To check the results, issue `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D udev=true`: This parameter enables support for using Udev instead of `mknod`.

`-D valgrind=disabled`: This parameter disables building libdrm with valgrind support. This fixes building some packages that use libdrm. Change this parameter to "enabled" (or remove it) if you need support for valgrind.

Contents

Installed Programs: None

Installed Libraries: libdrm_amdgpu.so, libdrm_intel.so, libdrm_nouveau.so, libdrm_radeon.so, and libdrm.so

Installed Directories: /usr/include/libdrm and /usr/share/libdrm

Short Descriptions

libdrm_amdgpu.so	contains the AMDGPU specific Direct Rendering Manager functions
libdrm_intel.so	contains the Intel specific Direct Rendering Manager functions
libdrm_nouveau.so	contains the open source nVidia (Nouveau) specific Direct Rendering Manager functions
libdrm_radeon.so	contains the AMD Radeon specific Direct Rendering Manager functions
libdrm.so	contains the Direct Rendering Manager API functions

libepoxy-1.5.10

Introduction to libepoxy

libepoxy is a library for handling OpenGL function pointer management.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libepoxy/1.5/libepoxy-1.5.10.tar.xz>
- Download MD5 sum: 10c635557904aed5239a4885a7c4efb7
- Download size: 220 KB
- Estimated disk space required: 13 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

libepoxy Dependencies

Required

[Mesa-24.1.5](#)

Optional

[Doxygen-1.12.0](#) (for documentation)

Installation of libepoxy

Install libepoxy by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D docs=true`: If you have [Doxygen-1.12.0](#) installed, add this option to generate additional documentation.

Contents

Installed Programs: None

Installed Libraries: libepoxy.so

Installed Directories: /usr/include/epoxy

Short Descriptions

libepoxy.so contains API functions for handling OpenGL function pointer management

libnotify-0.8.3

Introduction to libnotify

The libnotify library is used to send desktop notifications to a notification daemon, as defined in the Desktop Notifications spec. These notifications can be used to inform the user about an event or display some form of information without getting in the user's way.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libnotify/0.8/libnotify-0.8.3.tar.xz>
- Download MD5 sum: 952e55f45067b6a5b9eab4310c9cd658
- Download size: 104 KB
- Estimated disk space required: 2.3 MB
- Estimated build time: 0.1 SBU

libnotify Dependencies

Required

[GTK+-3.24.43](#)

Optional (Required if building GNOME)

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[Gi-DocGen-2024.1](#) and [xmlto-0.0.29](#)

Required (runtime)

[notification-daemon-3.20.0](#), [xfce4-notifyd-0.9.6](#), or [lxqt-notificationd-2.0.1](#)

Note

GNOME Shell and KDE KWin provide their own notification daemons.

Installation of libnotify

Install libnotify by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false    \
            -D man=false        \
            ..
            &&
ninja
```

This package does not come with a test suite.

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/docs_dir =/s@\$@ / 'libnotify'@" \
     -i ..../docs/reference/meson.build &&
meson configure -D gtk_doc=true &&
ninja
```

Now, as the `root` user:

```
ninja install &&
if [ -e /usr/share/doc/libnotify ]; then
  rm -rf /usr/share/doc/libnotify-0.8.3
  mv -v /usr/share/doc/libnotify{-,0.8.3}
fi
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: notify-send

Installed Library: libnotify.so

Installed Directories: /usr/include/libnotify and /usr/share/gtk-doc/html/libnotify

Short Descriptions

notify-send is a command used to send notifications
libnotify.so contains the libnotify API functions

libxklavier-5.4

Introduction to libxklavier

The libxklavier package contains a utility library for X keyboard.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://people.freedesktop.org/~svu/libxklavier-5.4.tar.bz2>
- Download MD5 sum: 13af74dcb6011ecedf1e3ed122bd31fa
- Download size: 384 KB
- Estimated disk space required: 5.2 MB
- Estimated build time: less than 0.1 SBU

libxklavier Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended), [ISO Codes-4.16.0](#), [libxml2-2.13.3](#) and [Xorg Libraries](#)

Optional

[GTK-Doc-1.34.0](#) and [Vala-0.56.17](#)

Installation of libxklavier

Install libxklavier by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

--enable-gtk-doc: This parameter is normally used if GTK-Doc is installed and you wish to rebuild and install the API documentation. It is broken for this package due to the use of a long deprecated gtk-doc program that is no longer available.

Contents

Installed Programs: None

Installed Library: libxklavier.so

Installed Directories: /usr/include/libxklavier and /usr/share/gtk-doc/html/libxklavier

Short Descriptions

libxklavier.so contains XKB utility functions

Pango-1.54.0

Introduction to Pango

Pango is a library for laying out and rendering text, with an emphasis on internationalization. It can be used anywhere that text layout is needed, though most of the work on Pango so far has been done in the context of the GTK+ widget toolkit.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/pango/1.54/pango-1.54.0.tar.xz>
- Download MD5 sum: 7ad89b03f6850e0be28f91522b793842
- Download size: 1.9 MB
- Estimated disk space required: 27 MB (with tests)
- Estimated build time: 0.1 SBU (Using parallelism=4; with tests)

Pango Dependencies

Required

[Fontconfig-2.15.0](#) (must be built with [FreeType-2.13.3](#) using [harfBuzz-9.0.0](#)), [FriBidi-1.0.15](#), and [GLib-2.80.4](#) (GObject Introspection required for GNOME)

Recommended

[Cairo-1.18.0](#) (built after [harfBuzz-9.0.0](#)) and [Xorg Libraries](#)

Optional

[Cantarell fonts](#) (for tests), [Gi-DocGen-2024.1](#) (to generate documentation), [help2man](#), [libthai](#), and [sysprof](#)

Installation of Pango

Install Pango by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr           \
            --buildtype=release    \
            --wrap-mode=nofallback \
            ..
            &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/docs_dir =/s@\$@ / 'pango-1.54.0'@" -i ..../docs/meson.build &&
meson configure -D documentation=true                                &&
ninja
```

To test the results, issue: `ninja test`. Two tests, testiter and test-font, are known to fail for unknown reason.

Now, as the `root` user:

```
ninja install
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

-D introspection=disabled: Use this switch if you do not want to use GObject Introspection.

--wrap-mode=nofallback: This switch prevents `meson` from using subproject fallbacks for any dependency declarations in the build files, stopping it downloading any optional dependency which is not installed on the system.

Configuring Pango

Config Files

`/etc/pango/pangorc`, `~/.pangorc` and the file specified in the environment variable `PANGO_RC_FILE`

Contents

Installed Programs: `pango-list`, `pango-segmentation`, and `pango-view`

Installed Libraries: `libpango-1.0.so`, `libpangocairo-1.0.so`, `libpangoft2-1.0.so`, and `libpangoft-1.0.so`

Installed Directories: `/usr/include/pango-1.0` and `/usr/share/doc/pango-1.54.0` (if `gi-docgen` is available)

Short Descriptions

<code>pango-list</code>	displays a list of fonts that Pango can use that are currently installed on the system
<code>pango-segmentation</code>	shows text segmentation as determined by Pango.
<code>pango-view</code>	renders a given text file through Pango for viewing purposes
<code>libpango-1.0.so</code>	contains low level layout rendering routines, a high level driver for laying out entire blocks of text, and routines to assist in editing internationalized text

Pangomm-2.46.4

Introduction to Pangomm

The Pangomm package provides a C++ interface to Pango.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/pangomm/2.46/pangomm-2.46.4.tar.xz>
- Download MD5 sum: 5947d35899db62813531d7ea8faee60c
- Download size: 680 KB
- Estimated disk space required: 8.9 MB
- Estimated build time: 0.2 SBU

Pangomm Dependencies

Required

[libcairomm-1.14.5](#), [GLibmm-2.66.7](#) and [Pango-1.54.0](#)

Installation of Pangomm

Install Pangomm by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Library: libpangomm-1.4.so

Installed Directories: /usr/include/pangomm-1.4, /usr/lib/pangomm-1.4, and /usr/share/{devhelp/books/pangomm-1.4, doc/pangomm-2.46.4}

Short Descriptions

libpangomm-1.4.so contains the Pango API classes

Pangomm-2.54.0

Introduction to Pangomm

The Pangomm package provides a C++ interface to Pango. This version is part of a new API for supporting gtkmm-4.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/pangomm/2.54/pangomm-2.54.0.tar.xz>
- Download MD5 sum: 19e0266fdd4b47d5fadd9f16ee5f728d
- Download size: 768 KB
- Estimated disk space required: 12 MB
- Estimated build time: 0.2 SBU

Pangomm Dependencies

Required

[libcairomm-1.18.0](#), [GLibmm-2.80.0](#) and [Pango-1.54.0](#)

Installation of Pangomm

Install Pangomm by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Library: libpangomm-2.48.so

Installed Directories: /usr/include/pangomm-2.48, /usr/lib/pangomm-2.48, and /usr/share/{devhelp/books/pangomm-2.48, doc/pangomm-2.46.4}

Short Descriptions

libpangomm-2.48.so contains the Pango API classes

qt5-components-5.15.14

Introduction to qt5 components

Qt5 is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) (in which cases qt5 is classified as a widget toolkit), and also used for developing non-GUI programs such as command-line tools and consoles for servers.

Important

The full qt5 package consists of 43 separate components. These instructions use a small subset of the full qt5 installation instructions. They cover the components needed to build packages in BLFS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.qt.io/archive/qt/5.15/5.15.14/single/qt-everywhere-opensource-src-5.15.14.tar.xz>
- Download MD5 sum: 0265e7988790d8845441558130d59311
- Download size: 630 MB
- Estimated disk space required: 9.0 GB (154 MB installed)
- Estimated build time: 7.5 SBU (using parallelism=4)

Additional Downloads

- Now that qt5 updates are restricted to commercial customers, upstream patches for the various modules are being curated at kde. Patches for the modules required by packages in BLFS have been aggregated for the non-modular qt5 build we use.

Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/qt-everywhere-opensource-src-5.15.14-kf5-1.patch>

Details of the kde curation can be found at <https://dot.kde.org/2021/04/06/announcing-kdes-qt-5-patch-collection> and <https://community.kde.org/Qt5PatchCollection>.

qt5 Dependencies

Required

[Xorg Libraries](#)

Recommended

[alsa-lib-1.2.12](#), [at-spi2-core-2.52.0](#), [Cups-2.4.10](#), [double-conversion-3.3.0](#), [GLib-2.80.4](#), [harfBuzz-9.0.0](#), [ICU-75.1](#), [hicolor-icon-theme-0.18](#), [Mesa-24.1.5](#), [libjpeg-turbo-3.0.1](#), [libxkbcommon-1.7.0](#), [SQLite-3.46.1](#), [Wayland-1.23.0](#) (Mesa must be built with Wayland EGL backend), [xcb-util-image-0.4.1](#), [xcb-util-keysyms-0.4.1](#), [xcb-util-renderutil-0.3.10](#), and [xcb-util-wm-0.4.2](#)

Optional

[GTK+-3.24.43](#), [libinput-1.26.1](#) [MariaDB-10.11.8](#) or [MySQL](#), [MIT Kerberos V5-1.21.3](#), [mtdev-1.1.7](#) [PostgreSQL-16.4](#), and [unixODBC-2.3.12](#)

Setting the installation prefix

Installing in /opt/qt5

The BLFS editors recommend installing qt5 in a directory other than `/usr`, i.e. `/opt/qt5`. To do this, set the following environment variable:

```
export QT5PREFIX=/opt/qt5
```

Tip

Sometimes, the installation paths are hardcoded into installed files. This is the reason why `/opt/qt5` is used as installation prefix instead of `/opt/qt-5.15.14`. To create a versioned qt5 directory after the installation is complete, you may rename the directory and create a symlink:

```
mkdir -pv /opt/qt-5.15.14  
ln -sfnv qt-5.15.14 /opt/qt5
```

Later on, you may want to install other versions of qt5. To do that, just remove the symlink and use `/opt/qt5` as the prefix again. Which version of qt5 you use depends only on where the symlink points.

Installation of qt5

Warning

If qt5 is being reinstalled into the same directory as an existing instance, run the commands done by `root`, such as `make install`, from a console or non-qt5 based window manager. It overwrites qt5 libraries that should not be in use during the install process.

Caution

If you did not install some of the recommended dependencies, examine `./configure --help` output to check how to disable them or use internal versions bundled in the source tarball.

Note

The BLFS editors do not recommend installing qt5 into the `/usr` hierarchy because it becomes difficult to find components and to update to a new version. If you do want to install qt5 in `/usr`, the directories need to be specified explicitly. In this case, set `QT5PREFIX=/usr` and add the following to the configure arguments below:

```
-archdatadir    /usr/lib/qt5      \  
-bindir        /usr/bin          \  
-plugindir     /usr/lib/qt5/plugins \  
-importdir     /usr/lib/qt5/imports \  
-headerdir     /usr/include/qt5   \  
-datadir       /usr/share/qt5    \  
-docdir        /usr/share/doc/qt5 \  
-translationdir /usr/share/qt5/translations \  
-examplesdir   /usr/share/doc/qt5/examples
```

First, apply a patch to pull in the fixes curated by KDE for those modules which are required by packages mentioned in this book:

```
patch -Np1 -i .../qt-everywhere-opensource-src-5.15.14-kf5-1.patch
```

The patch is supposed to be used in a git repository. If it isn't, some files that should be modified by `configure` after applying the patch are not touched. Fortunately, it is enough to create a `.git` directory in the `qmake` directory where the `configure` script is run:

```
mkdir -pv qtbase/.git
```

At this point we want to set up skipping most components. Do that with:

```
ls -Fd qt* | grep / | sed 's/^/-skip /;s@/@@' > tempconf &&  
sed -i -r '/base|tools|x11extras|svg|declarative|wayland/d' tempconf
```

If you want to install the [NetworkManager-1.48.8](#) examples, run: `sed -i '/qtdoc/d' tempconf`.

If you choose to not install [Qt-6.7.2](#) and want to use qt5 with wireshark, run: `sed -i '/qtmultimedia/d' tempconf`.

Now, install qt5 by running the following commands:

```
./configure -prefix $QT5PREFIX  \  
-sysconfdir /etc/xdg \  
-
```

```

    -confirm-license      \
    -opensource          \
    -dbus-linked         \
    -openssl-linked     \
    -system-harfbuzz    \
    -system-sqlite       \
    -nomake examples    \
    -no-rpath            \
    -journald           \
    $(cat tempconf)     &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Remove references to the build directory from installed library dependency (prl) files by running the following command as `root` user:

```
find $QT5PREFIX/ -name \*.prl \
-exec sed -i -e '/^QMAKE_PRL_BUILD_DIR/d' {} \;
```

Install images and create the menu entries for installed applications. The `QT5BINDIR` variable is used here to point to the directory for the executable programs. If you have changed the bindir above, `QT5BINDIR` will need to be adjusted below. Be sure that the `QT5BINDIR` variable is defined in root's environment and as the `root` user:

```

QT5BINDIR=$QT5PREFIX/bin

install -v -dm755 /usr/share/pixmaps/ &&

install -v -Dm644 qttools/src/assistant/assistant/images/assistant-128.png \
/usr/share/pixmaps/assistant-qt5.png &&

install -v -Dm644 qttools/src/designer/src/designer/images/designer.png \
/usr/share/pixmaps/designer-qt5.png &&

install -v -Dm644 qttools/src/linguist/linguist/images/icons/linguist-128-32.png \
/usr/share/pixmaps/linguist-qt5.png &&

install -v -Dm644 qttools/src/qdbus/qdbusviewer/images/qdbusviewer-128.png \
/usr/share/pixmaps/qdbusviewer-qt5.png &&

install -dm755 /usr/share/applications &&

cat > /usr/share/applications/assistant-qt5.desktop << EOF
[Desktop Entry]
Name=Qt5 Assistant
Comment=Shows Qt5 documentation and examples
Exec=$QT5BINDIR/assistant
Icon=assistant-qt5.png
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;Documentation;
EOF

cat > /usr/share/applications/designer-qt5.desktop << EOF
[Desktop Entry]
Name=Qt5 Designer
GenericName=Interface Designer
Comment=Design GUIs for Qt5 applications
Exec=$QT5BINDIR/designer
Icon=designer-qt5.png
MimeType=application/x-designer;
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;
EOF

cat > /usr/share/applications/linguist-qt5.desktop << EOF
[Desktop Entry]
```

```

Name=Qt5 Linguist
Comment>Add translations to Qt5 applications
Exec=$QT5BINDIR/linguist
Icon=linguist-qt5.png
MimeType=text/vnd.trolltech.linguist;application/x-linguist;
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;
EOF

cat > /usr/share/applications/qdbusviewer-qt5.desktop << EOF
[Desktop Entry]
Name=Qt5 QDBusViewer
GenericName=D-Bus Debugger
Comment=Debug D-Bus applications
Exec=$QT5BINDIR/qdbusviewer
Icon=qdbusviewer-qt5.png
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;Debugger;
EOF

```

Some packages such as [VLC-3.0.21](#) look for certain executables with a -qt5 suffix. Run the following command as the *root* user to create the necessary symlinks:

```

for file in moc uic rcc qmake lconvert lrelease lupdate; do
    ln -sfvn $QT5BINDIR/$file /usr/bin/$file-qt5
done

```

Command Explanations

-confirm-license: Accept license without prompting user during configuration.

-opensource: Install the opensource version of qt.

-nomake examples: This switch disables building of the example programs included in the source tarball. Remove it if you want to build them.

-system-sqlite: This switch enables use of the system version of SQLite.

-system-harfbuzz: This switch enables use of the system version of Harfbuzz.

-dbus-linked -openssl-linked: These switches enable explicit linking of the D-Bus and OpenSSL libraries into qt5 libraries instead of `dlopen()`-ing them.

-journald: This switch allows to send qt messages to the `journald` logging system.

`$ (cat tempconf)`: This command provides a list of components that should not be built. These entries are in the form of "`-skip <component>`".

Configuring qt5

Configuration Information

If [Sudo-1.9.15p5](#) is installed, QT5DIR should be available to the super user as well. Execute the following commands as the *root* user:

```

cat > /etc/sudoers.d/qt << "EOF"
Defaults env_keep += QT5DIR
EOF

```

If you installed qt5 in /usr

If you installed qt5 in `/usr`, create an environment variable needed by certain packages. As the *root* user:

```

cat > /etc/profile.d/qt5.sh << "EOF"
# Begin /etc/profile.d/qt5.sh

QT5DIR=/usr

```

```

export QT5DIR

# End /etc/profile.d/qt5.sh
EOF

```

If you did not install qt5 in /usr

If you installed qt5 in a location other than `/usr`, you need to update the following configuration files so that qt5 is correctly found by other packages and system processes.

As the `root` user, update the `/etc/ld.so.conf` file and the dynamic linker's run-time cache file:

```

cat >> /etc/ld.so.conf << EOF
# Begin Qt addition

/opt/qt5/lib

# End Qt addition
EOF

ldconfig

```

As the `root` user, create the `/etc/profile.d/qt5.sh` file:

```

cat > /etc/profile.d/qt_5.sh << "EOF"
# Begin /etc/profile.d/qt_5.sh

QT5DIR=/opt/qt5

pathappend $QT5DIR/bin PATH
pathappend $QT5DIR/lib/pkgconfig PKG_CONFIG_PATH

export QT5DIR

# End /etc/profile.d/qt_5.sh
EOF

```

Contents

Installed Programs: `assistant`, `designer`, `fixqt4headers.pl`, `Iconvert`, `linguist`, `Iprodump`, `Irelease`, `Irelease-pro`, `lupdate`, `lupdate-pro`, `moc`, `pixeltool`, `qcollectiongenerator`, `qdbus`, `qdbuscpp2xml`, `qdbusviewer`, `qdbusxml2cpp`, `qdistancefieldgenerator`, `qdoc`, `qhelpgenerator`, `qlalr`, `qmake`, `qml`, `qmlcachegeen`, `qmleasing`, `qmlformat`, `qmlimportscanner`, `qmllint`, `qmllmin`, `qmlplugindump`, `qmlpreview`, `qmlprofiler`, `qmlscene`, `qmltestrunner`, `qmftime`, `qmlyperegistrar`, `qtattributionscanner`, `qtdiag`, `qtpaths`, `qtplugininfo`, `qtwaylandscanner`, `qvkgen`, `rcc`, `syncqt.pl`, `tracegen`, and `uic`

Installed Libraries: `libQt5AccessibilitySupport.a`, `libQt5Bootstrap.a`, `libQt5Concurrent.so`, `libQt5Core.so`, `libQt5DBus.so`, `libQt5Designer.so`, `libQt5DesignerComponents.so`, `libQt5DeviceDiscoverySupport.a`, `libQt5EddisSupport.a`, `libQt5EglFSDDeviceIntegration.so`, `libQt5EglFsKmsSupport.so`, `libQt5EglSupport.a`, `libQt5EventDispatcherSupport.a`, `libQt5FbSupport.a`, `libQt5FontDatabaseSupport.a`, `libQt5GlxSupport.a`, `libQt5Gui.so`, `libQt5Help.so`, `libQt5InputSupport.a`, `libQt5KmsSupport.a`, `libQt5LinuxAccessibilitySupport.a`, `libQt5Network.so`, `libQt5OpenGL.so`, `libQt5OpenGLExtensions.a`, `libQt5PacketProtocol.a`, `libQt5PlatformCompositorSupport.a`, `libQt5PrintSupport.so`, `libQt5Qml.so`, `libQt5QmlDebug.a`, `libQt5QmlDevTools.a`, `libQt5QmlModels.so`, `libQt5QmlWorkerScript.so`, `libQt5Quick.so`, `libQt5QuickParticles.so`, `libQt5QuickShapes.so`, `libQt5QuickTest.so`, `libQt5QuickWidgets.so`, `libQt5ServiceSupport.a`, `libQt5Sql.so`, `libQt5Svg.so`, `libQt5Test.so`, `libQt5ThemeSupport.a`, `libQt5UiTools.a`, `libQt5WaylandClient.so`, `libQt5WaylandCompositor.so`, `libQt5Widgets.so`, `libQt5X11Extras.so`, `libQt5XcbQpa.so`, `libQt5XkbCommonSupport.a`, `libQt5Xml.so`, and several plugins under `/opt/qt5/{plugins,qml}`

Installed Directories: `/usr/include/qt5`, `/usr/lib/qt5`, `/usr/share/doc/qt5`, and `/usr/share/qt5` OR `/opt/qt5` and `/opt/qt-5.15.14`

Short Descriptions

Short Descriptions

<code>assistant</code>	is a tool for presenting on-line documentation
<code>designer</code>	is a full-fledged GUI builder. It includes powerful features such as preview mode, automatic widget layout, support for custom widgets, and an advanced property editor
<code>fixqt4headers.pl</code>	is a script to replace all Qt 4 style includes with Qt 5 includes
<code>Iconvert</code>	is part of Qt5's Linguist tool chain. It can be used as a standalone tool to convert and filter translation data files

<code>linguist</code>	provides support for translating applications into local languages
<code>lrelease</code>	is a simple command line tool. It reads XML-based translation file in TS format and produces message files used by the application
<code>lrelease-pro</code>	extracts project information from qmake projects and passes it to lrelease
<code>lupdate</code>	finds the translatable strings in the specified source, header and Qt Designer interface files, and stores the extracted messages in translation files to be processed by lrelease
<code>lupdate-pro</code>	reads a Qt project file and passes the collected information to lupdate
<code>moc</code>	generates Qt meta object support code
<code>pixeltool</code>	is a desktop magnifier - as you move your mouse around the screen, it will show the magnified contents in its window
<code>qcollectiongenerator</code>	processes and converts Qt help files
<code>qdbuscpp2xml</code>	takes a C++ source file and generates a D-Bus XML definition of the interface
<code>qdbus</code>	lists available services, object paths, methods, signals, and properties of objects on a bus
<code>qdbusviewer</code>	is a graphical D-Bus browser
<code>qdbusxml2cpp</code>	is a tool that can be used to parse interface descriptions and produce static code representing those interfaces
<code>qdistancefieldgenerator</code>	allows a font cache to be pregenerated for Text.QtRendering to speed up an application's startup if the user interface has a lot of text, or multiple fonts, or a large amount of distinct characters, e.g. in CJK writing systems
<code>qdoc</code>	is a tool used by Qt Developers to generate documentation for software projects
<code>qhelpgenerator</code>	is a tool used to generate a Qt compressed help file
<code>qlalr</code>	is a tool used to generate code from grammar specifications
<code>qmake</code>	uses information stored in project files to determine what should go in the makefiles it generates
<code>qml</code>	executes a QML file
<code>qmlcachegen</code>	supports the creation of cache files at build time
<code>qmleasing</code>	is a tool used to define the easing curves using an interactive curve editor
<code>qmlformat</code>	formats QML files according to the QML coding conventions
<code>qmlimportscanner</code>	is a tool used to import QML files from a directory
<code>qmllint</code>	is a syntax checker for QML files
<code>qmlmin</code>	removes comments and layout characters from a QML file
<code>qmlplugindump</code>	is a tool to create a qmltypes file
<code>qmlpreview</code>	is a tool used to analyze QML applications
<code>qmlprofiler</code>	is a tool which watches QML and JavaScript files on disk and updates the application live with any changes
<code>qmlscene</code>	is a utility that loads and displays QML documents even before the application is complete
<code>qmltestrunner</code>	is a tool used to make tests
<code>qtattributionsscanner</code>	processes qt_attribution.json files in Qt sources
<code>qtdiag</code>	is a tool for reporting diagnostic information about Qt and its environment
<code>qtpaths</code>	is a tool to query Qt path information
<code>qtplugininfo</code>	dumps meta-data about Qt plugins in JSON format
<code>qtwaylandscanner</code>	converts Wayland specfiles to C++ headers and code needed for QtWayland
<code>qvkgend</code>	converts Vulkan specfiles to C++ headers and code
<code>rcc</code>	is a resource compiler used in conjunction with designer
<code>syncqt.pl</code>	is a script to create the forwarding headers in the include directories. It is an internal development tool
<code>tracegen</code>	is a trace generator for LTTng or ETW
<code>uic</code>	is a Qt user interface compiler

Qt-6.7.2

Introduction to Qt6

Qt6 is a cross-platform application framework that is widely used for developing application software with a graphical user interface (GUI) (in which cases Qt6 is classified as a widget toolkit), and also used for developing non-GUI programs such as command-line tools and consoles for servers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.qt.io/archive/qt/6.7/6.7.2/single/qt-everywhere-src-6.7.2.tar.xz>
- Download MD5 sum: 06d35b47349c7c0a45710daad359e07b
- Download size: 892 MB
- Estimated disk space required: 37 GB (376 MB installed)
- Estimated build time: 12 SBU (using parallelism=8)

Qt6 Dependencies

Required

[Xorg Libraries](#)

Recommended

[alsa-lib-1.2.12](#), [make-ca-1.14](#), [Cups-2.4.10](#), [double-conversion-3.3.0](#), [GLib-2.80.4](#), [gst-plugins-base-1.24.7](#) (QtMultimedia backend), [harfBuzz-9.0.0](#), [ICU-75.1](#), [JasPer-4.2.4](#), [libjpeg-turbo-3.0.1](#), [libinput-1.26.1](#), [libmng-2.0.3](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [libxkbcommon-1.7.0](#), [Mesa-24.1.5](#), [mtdev-1.1.7](#), [pcre2-10.44](#), [SQLite-3.46.1](#), [Wayland-1.23.0](#) (Mesa must be built with Wayland EGL backend), [xcb-util-cursor-0.1.4](#), [xcb-util-image-0.4.1](#), [xcb-util-keysyms-0.4.1](#), [xcb-util-renderutil-0.3.10](#), and [xcb-util-wm-0.4.2](#)

Optional

[BlueZ-5.77](#) (for sdpscanner, and at runtime for QtConnectivity module), [GTK+-3.24.43](#), [ibus-1.5.30](#), [MariaDB-10.11.8](#) or [MySQL](#), [MIT Kerberos V5-1.21.3](#), [pciutils-3.13.0](#), [PostgreSQL-16.4](#), [PulseAudio-17.0](#), [SDL2-2.30.6](#), [unixODBC-2.3.12](#), [assimp](#), [Flite](#), [Firebird](#), [FreeTDS](#), [libproxy](#), [OpenAL](#), [speech-dispatcher](#), and [tslib](#)

Setting the installation prefix

Installing in /opt/qt6

The BLFS editors recommend installing Qt6 in a directory other than `/usr`, i.e. `/opt/qt6`. This is needed to avoid conflicts with Qt5. To do this, set the following environment variable:

```
export QT6PREFIX=/opt/qt6
```

Tip

Sometimes the installation paths are hardcoded into installed files. This is the reason why `/opt/qt6` is used as installation prefix instead of `/opt/qt-6.7.2`. To create a versioned Qt6 directory, you may rename the directory and create a symlink:

```
mkdir -pv /opt/qt-6.7.2
ln -sfnv qt-6.7.2 /opt/qt6
```

Later on, you may want to install other versions of Qt6. To do that, just remove the symlink, create the new versioned directory, and recreate the `/opt/qt6` symlink again before building the new version. Which version of Qt6 you use depends only on where the symlink points.

Note

If [qca-2.3.9](#) has been installed and you are reinstalling or updating this package, then qca will need to be reinstalled.

Installation of Qt6

Warning

If Qt6 is being reinstalled into the same directory as an existing instance, run the commands done by `root`, such as `make install`, from a console or non-Qt6 based window manager. It overwrites Qt6 libraries that should not be in use during the install process.

Caution

If you did not install some of the recommended dependencies, examine `./configure --help` output to check how to disable them or use internal versions bundled in the source tarball.

Note

The build time and space required for the full Qt6 is quite long. The instructions below do not build the tutorials and examples. Removing the `-nomake` line will create additional resources..

Note

The BLFS editors do not recommend installing Qt6 into the `/usr` hierarchy because it becomes difficult to find components and to update to a new version. There are also some programs that conflict with those installed by Qt5.

Disable a conflicting declaration on i686 systems:

```
if [ "$(uname -m)" == "i686" ]; then
    sed -e "/^#elif defined(Q_CC_GNU_ONLY)/s/.*/&\& 0/" \
        -i qtbase/src/corelib/global/qtypes.h
fi
```

Install Qt6 by running the following commands:

```
./configure -prefix $QT6PREFIX \
            \
            -sysconfdir /etc/xdg \
            \
            -dbus-linked \
            \
            -openssl-linked \
            \
            -system-sqlite \
            \
            -nomake examples \
            \
            -no-rpath \
            \
            -journald \
            \
            -skip qt3d \
            \
            -skip qtquick3dphysics \
            \
            -skip qtwebengine \
            \
            -W no-dev \
            \
            &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Remove references to the build directory from installed library dependency (prl) files by running the following command as the `root` user:

```
find $QT6PREFIX/ -name *.prl \
    -exec sed -i -e '/^QMAKE_PRL_BUILD_DIR/d' {} \;
```

Install images and create the menu entries for installed applications. Again as the `root` user:

```
pushd qttools/src &&
install -v -Dm644 assistant/assistant/images/assistant-128.png \
        /usr/share/pixmaps/assistant-qt6.png &&
install -v -Dm644 designer/src/designer/images/designer.png \
        /usr/share/pixmaps/designer-qt6.png &&
```

```

install -v -Dm644 linguist/linguist/images/icons/linguist-128-32.png \
        /usr/share/pixmaps/linguist-qt6.png           &&
install -v -Dm644 qdbus/qdbusviewer/images/qdbusviewer-128.png \
        /usr/share/pixmaps/qdbusviewer-qt6.png         &&
popd &&

cat > /usr/share/applications/assistant-qt6.desktop << EOF
[Desktop Entry]
Name=Qt6 Assistant
Comment=Shows Qt6 documentation and examples
Exec=$QT6PREFIX/bin/assistant
Icon=assistant-qt6.png
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;Documentation;
EOF

cat > /usr/share/applications/designer-qt6.desktop << EOF
[Desktop Entry]
Name=Qt6 Designer
GenericName=Interface Designer
Comment=Design GUIs for Qt6 applications
Exec=$QT6PREFIX/bin/designer
Icon=designer-qt6.png
MimeType=application/x-designer;
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;
EOF

cat > /usr/share/applications/linguist-qt6.desktop << EOF
[Desktop Entry]
Name=Qt6 Linguist
Comment>Add translations to Qt6 applications
Exec=$QT6PREFIX/bin/linguist
Icon=linguist-qt6.png
MimeType=text/vnd.trolltech.linguist;application/x-linguist;
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;
EOF

cat > /usr/share/applications/qdbusviewer-qt6.desktop << EOF
[Desktop Entry]
Name=Qt6 QDBusViewer
GenericName=D-Bus Debugger
Comment=Debug D-Bus applications
Exec=$QT6PREFIX/bin/qdbusviewer
Icon=qdbusviewer-qt6.png
Terminal=false
Encoding=UTF-8
Type=Application
Categories=Qt;Development;Debugger;
EOF

```

Command Explanations

-nomake examples: This switch disables building of the example programs included in the source tarball. Remove it if you want to build them.

-skip qt3d: This switch disables building qt3d support. There is a problem building these files without an external library and no packages in BLFS use qt3d.

-system-sqlite: This switch enables use of the system version of SQLite.

-dbus-linked -openssl-linked: These switches enable explicit linking of the D-Bus and OpenSSL libraries into Qt6 libraries instead of `dlopen()`-ing them.

-journalid: This switch allows sending Qt messages to the `journald` logging system.

`-skip qtwebengine`: This switch disables building the QtWebEngine. The BLFS editors have chosen to build [qtwebengine-6.7.2](#) separately.

`-skip qtquick3dphysics`: This switch disables building the Qt Quick 3D Physics submodule. On 32-bit systems, this will cause the build process to fail with an inlining error in Qt6's bundled copy of the PhysX SDK.

Configuring Qt6

Configuration Information

If [Sudo-1.9.15p5](#) is installed, QT6DIR should be available to the super user as well. Execute the following commands as the `root` user:

```
cat > /etc/sudoers.d/qt << "EOF"
Defaults env_keep += QT6DIR
EOF
```

You now need to update the following configuration files so that Qt6 is correctly found by other packages and system processes.

As the `root` user, update the `/etc/ld.so.conf` file and the dynamic linker's run-time cache file:

```
cat >> /etc/ld.so.conf << EOF
# Begin Qt addition

/opt/qt6/lib

# End Qt addition
EOF

ldconfig
```

As the `root` user, create the `/etc/profile.d/qt6.sh` file:

```
cat > /etc/profile.d/qt6.sh << "EOF"
# Begin /etc/profile.d/qt6.sh

QT6DIR=/opt/qt6

pathappend $QT6DIR/bin          PATH
pathappend $QT6DIR/lib/pkgconfig PKG_CONFIG_PATH

export QT6DIR

# End /etc/profile.d/qt6.sh
EOF
```

Contents

Installed Programs: androiddeployqt (hard link to androiddeployqt6), androiddeployqt6, androidtestrunner, assistant, balsam, balsamui, canbusutil, cooker, designer, instancer, lconvert, linguist, lrelease, lupdate, materialeditor, meshdebug, pixeltool, qdbus, qdbuscpp2xml, qdbusviewer, qdbusxml2cpp, qdistancefieldgenerator, qdoc, qmake (hard link to qmake6), qmake6, qml, qmldom, qmleasing, qmlformat, qmllint, qmls, qmlplugindump, qmlpreview, qmlprofiler, qmlscene, qmltc, qmltestrunner, qmltime, qqem, qsb, qtdiag (hard link to qtdiag6), qtdiag6, qtpaths (hard link to qtpaths6), qtpaths6, qtplugininfo, qt-cmake, qt-cmake-create, qt-configure-module, shadergen, and shapegen

Installed Libraries: libQt6Bluetooth.so, libQt6Bodymovin.so, libQt6BundledEmbree.a, libQt6BundledPhysX.a, libQt6BundledResonanceAudio.a, libQt6Charts.so, libQt6ChartsQml.so, libQt6Coap.so, libQt6Concurrent.so, libQt6Core.so, libQt6Core5Compat.so, libQt6DBus.so, libQt6DataVisualization.so, libQt6DataVisualizationQml.so, libQt6DeclarativeOpcua.so, libQt6Designer.so, libQt6DesignerComponents.so, libQt6DeviceDiscoverySupport.a, libQt6EglFSDDeviceIntegration.so, libQt6EglFksKmsGbmSupport.so, libQt6EglFksSupport.so, libQt6ExampleIcons.a, libQt6FbSupport.a, libQt6Graphs.so, libQt6Grpc.so, libQt6Gui.so, libQt6Help.so, libQt6HttpServer.so, libQt6InputSupport.a, libQt6JsonRpc.so, libQt6KmsSupport.a, libQt6LabsAnimation.so, libQt6LabsFolderListModel.so, libQt6LabsQmlModels.so, libQt6LabsSettings.so, libQt6LabsSharedImage.so, libQt6LabsWavefrontMesh.so, libQt6LanguageServer.so, libQt6Location.so, libQt6Mqtt.so, libQt6Multimedia.so, libQt6MultimediaQuick.so, libQt6MultimediaWidgets.so, libQt6Network.so, libQt6NetworkAuth.so, libQt6Nfc.so, libQt6OpcUa.so, libQt6OpenGL.so, libQt6OpenGLWidgets.so, libQt6PacketProtocol.a, libQt6Positioning.so, libQt6PositioningQuick.so, libQt6PrintSupport.so, libQt6Protobuf.so, libQt6Qml.so, libQt6QmlCompiler.so, libQt6QmlCore.so, libQt6QmlDebug.a, libQt6QmlDom.a, libQt6QmlLocalStorage.so, libQt6QmlILS.a, libQt6QmlModels.so, libQt6QmlToolingSettings.a, libQt6QmlTypeRegistrar.a, libQt6QmlWorkerScript.so,

libQt6QmlXmlListModel.so, libQt6Quick.so, libQt6Quick3D.so, libQt6Quick3DAssetImport.so, libQt6Quick3DAssetUtils.so, libQt6Quick3DEffects.so, libQt6Quick3DGlsParser.so, libQt6Quick3DHelpers.so, libQt6Quick3DHelpersImpl.so, libQt6Quick3DIblBaker.so, libQt6Quick3DParticleEffects.so, libQt6Quick3DParticles.so, libQt6Quick3DPhysics.so, libQt6Quick3DPhysicsHelpers.so, libQt6Quick3DRuntimeRender.so, libQt6Quick3DSpatialAudio.so, libQt6Quick3DUtils.so, libQt6QuickControls2.so, libQt6QuickControls2Impl.so, libQt6QuickControlsTestUtils.a, libQt6QuickDialogs2.so, libQt6QuickDialogs2QuickImpl.so, libQt6QuickDialogs2Utils.so, libQt6QuickEffects.so, libQt6QuickLayouts.so, libQt6QuickParticles.so, libQt6QuickShapes.so, libQt6QuickTemplates2.so, libQt6QuickTest.so, libQt6QuickTestUtils.a, libQt6QuickTimeline.so, libQt6QuickWidgets.so, libQt6RemoteObjects.so, libQt6RemoteObjectsQml.so, libQt6Sxml.so, libQt6SxmlQml.so, libQt6Sensors.so, libQt6SensorsQuick.so, libQt6SerialBus.so, libQt6SerialPort.so, libQt6ShaderTools.so, libQt6SpatialAudio.so, libQt6Sql.so, libQt6StateMachine.so, libQt6StateMachineQml.so, libQt6Svg.so, libQt6SvgWidgets.so, libQt6TextToSpeech.so, libQt6UiTools.so, libQt6VirtualKeyboard.so, libQt6WaylandClient.so, libQt6WaylandCompositor.so, libQt6WaylandEglClientHwIntegration.so, libQt6WaylandEglCompositorHwIntegration.so, libQt6WebChannel.so, libQt6WebChannelQuick.so, libQt6WebSockets.so, libQt6WebView.so, libQt6WebViewQuick.so, libQt6Widgets.so, libQt6WIShellIntegration.so, libQt6XcbQpa.so, and libQt6Xml.so

Installed Directories: /opt/qt6 and /opt/qt-6.7.2

Short Descriptions

assistant	is a tool for presenting on-line documentation
balsam	is a tool to convert 3D scenes from various creation tools to QML format, for use by the new QtQuick 3D library
canbusutil	is a tool to deal with arbitrary CAN bus frames. A Controller Area Network (CAN) is a vehicle bus standard designed to allow microcontrollers and devices to communicate with each other in applications without a host computer
designer	is a full-fledged GUI builder. It includes powerful features such as preview mode, automatic widget layout, support for custom widgets, and an advanced property editor
lconvert	is part of Qt6's Linguist tool chain. It can be used as a standalone tool to convert and filter translation data files
linguist	provides support for translating applications into local languages
lrelease	is a simple command line tool. It reads XML-based translation files in TS format and produces message files used by the application
lupdate	finds the translatable strings in the specified source, header and Qt Designer interface files, and stores the extracted messages in translation files to be processed by lrelease
meshdebug	displays information about qtquick-3d mesh files
pixeltool	is a desktop magnifier - as you move your mouse around the screen, it will show the magnified contents in its window
qdbus	lists available services, object paths, methods, signals, and properties of objects on a bus
qdbuscpp2xml	takes a C++ source file and generates a D-Bus XML definition of the interface
qdbusviewer	is a graphical D-Bus browser
qdbusxml2cpp	is a tool that can be used to parse interface descriptions and produce static code representing those interfaces,
qdistancefieldgenerator	allows a font cache to be pregenerated for Text.QtRendering to speed up an application's startup if the user interface has a lot of text, or multiple fonts, or a large amount of distinct characters, e.g. in CJK writing systems
qdoc	is a tool used by Qt Developers to generate documentation for software projects
qmake	uses information stored in project files to determine what should go in the makefiles it generates
qml	executes a QML file
qmleasing	is a tool used to define the easing curves using an interactive curve editor
qmlformat	formats QML files according to the QML coding conventions
qmllint	is a syntax checker for QML files
qmlplugindump	is a tool to create a qmltypes file
qmlpreview	is a tool used to analyze QML applications
qmlprofiler	is a tool which watches QML and JavaScript files on disk and updates the application live with any changes
qmlscene	is a utility that loads and displays QML documents even before the application is complete
qmltestrunner	is a tool used to make tests
qtdiag	is a tool for reporting diagnostic information about Qt and its environment

```
qtpaths          is a tool to query Qt path information  
qtplugininfo    dumps meta-data about Qt plugins in JSON format
```

QtWebEngine-6.7.2

Introduction to QtWebEngine

QtWebEngine integrates chromium's web capabilities into Qt. It ships with its own copy of ninja which it uses for the build if it cannot find a system copy, and various copies of libraries from ffmpeg, icu, libvpx, and zlib (including libminizip) which have been forked by the chromium developers.

This package and browsers using it may be useful if you need to use a website designed for google chrome, or chromium, browsers.

Warning

QtWebEngine uses a forked copy of chromium, and is therefore vulnerable to many issues found there. The Qt developers seem to fork a newer version for minor Qt versions, but because chromium moves to newer versions very often, by the time the Qt developers get a forked version to pass their extended tests it is always an old version and security fixes from chromium (some of which have a CVE number) can take several months to appear in a QtWebengine release, even if the severity has been rated as Critical.

Therefore, you should be wary of using QtWebEngine in a sensitive context and should always update to the next release as soon as it appears in this book, even if it is not flagged as a Security Update. Identifying which vulnerabilities have been fixed in a particular release requires pulling the appropriate 'based-NNN' branch just before the previous and current releases and is often impractical. Reports of fixed QTBUG items do not seem to be available and there is not any documentation in the tarball for changes after the qt-5 versions.

This package is known to build and work properly using an LFS 12.2 platform.

Warning

By default, ninja will use all online CPUs +2 (if at least 4 exist), even if they are not available to the current task because the build terminal has been restricted with 'taskset'. In BLFS, this package takes more time to build than any other. In one example, the build of this package crashed at about the 90 percent point due to an out of memory problem on a system with 24 cores and 32 GB of memory.

To work around this, see the Command Explanations below.

Package Information

- Download (HTTP): https://download.qt.io/official_releases/qt/6.7/6.7.2/submodules/qtwebengine-everywhere-src-6.7.2.tar.xz
- Download MD5 sum: 650bacb77f1fa3d4843e0667df07a1b8
- Download size: 525 MB
- Estimated disk space required: 8.7 GB (335 MB installed)
- Estimated build time: 40 SBU (Using parallelism=8)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/qtwebengine-6.7.2-ffmpeg7_fixes-1.patch

qtwebengine Dependencies

Required

[Cups-2.4.10](#), [html5lib-1.1](#), [nodejs-20.16.0](#), [nss-3.103](#), [pciutils-3.13.0](#), and [Qt-6.7.2](#)

Recommended

Note

If these packages are not installed, the build process will compile and install its own (perhaps older) version, with the side effect of increasing build and installed disk space and build time.

either [alsa-lib-1.2.12](#) or [PulseAudio-17.0](#) (or both), [FFmpeg-7.0.2](#), [ICU-75.1](#), [libwebp-1.4.0](#), [libxslt-1.1.42](#), and [Opus-1.5.2](#)

Optional

[libevent-2.1.12](#), [MIT Kerberos V5-1.21.3](#), [pipewire-1.2.3](#), [Poppler-24.08.0](#), [jsoncpp](#), [libsrtplib](#), [snappy](#)

Installation of qtwebengine

First, adapt the bundled copy of Chromium to the latest version of ffmpeg:

```
patch -Np1 -d src/3rdparty/chromium < ../../qtwebengine-6.7.2-ffmpeg7_fixes-1.patch
```

Next, fix a build failure that occurs because the BLFS build of ffmpeg does not use OpenH264:

```
sed -e '189 s/=/& false/' \
     -e '190 d' \
     -i.orig src/3rdparty/chromium/third_party/webrtc/webrtc.gni
```

Install qtwebengine by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_MESSAGE_LOG_LEVEL=STATUS \
      -D QT_FEATURE_webengine_system_ffmpeg=ON \
      -D QT_FEATURE_webengine_system_icu=ON \
      -D QT_FEATURE_webengine_system_libevent=ON \
      -D QT_FEATURE_webengine_proprietary_codecs=ON \
      -D QT_FEATURE_webengine_webrtc_pipewire=ON \
      -D QT_BUILD_EXAMPLES_BY_DEFAULT=OFF \
      -G Ninja ... &&

ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`CMAKE_MESSAGE_LOG_LEVEL=STATUS`: Output interesting messages that project users might be interested in. Ideally these should be concise, no more than a single line, but still informative.

`QT_FEATURE_webengine_system_*`: Specify what external packages the system should use.

`QT_BUILD_EXAMPLES_BY_DEFAULT=OFF`: Do not build examples by default.

`NINJAJOBS=4 make`: If you patched system ninja in LFS to recognize the NINJAJOBS environment variable, this command will run system ninja with the specified number of jobs (i.e. 4). There are several reasons why you might want to use options like this this:

- Building on a subset of CPUs allows measuring the build time for a smaller number of processors, and/or running other CPU-intensive tasks at the same time. For an editor on a machine with a lot of CPUs, trying to measure the build time for a 4-CPU machine, `NINJAJOBS=4 make` will give a reasonable approximation (there is a short period where N+2 python and node jobs run).
- On a machine with only 4 CPUs online, the default of scheduling N+2 jobs for qtwebengine is slower by between 3% and 7%, probably because of the size of the C++ files and their many includes and templates. Therefore, if in doubt set NINJAJOBS to the number of CPUs.
- Reducing the number of cores being used on long running, CPU intensive packages may alleviate heat problems.
- Reducing the number of cores will prevent potential out-of-memory problems on systems that do not have enough memory (or swap) when all cores are active. A suggested approach is to limit the number of cores to about one core for each 1.5 GB of combined RAM and swap space.

Configuring QtWebEngine

Configuration Information

If you are upgrading from an older minor version of this application, for some webpages to load you may need to clear the browser caches, e.g. for falkon they will be found in `~/.cache/falkon/`. You will need to do this if the browser starts to render the page and then changes to a blank tab with a message that something went wrong, and a button to Retry. Even after removing the old caches, you may need to retry a few times for each affected tab.

If a browser using this package fails to run and when run from a terminal it reports 'Trace/breakpoint trap' that is probably a kernel configuration issue - there is no need to rebuild QtWebEngine, see the next section, recompile the kernel and reboot to the new kernel.

Kernel Configuration

This package does not require any of the optional kernel namespace items, but if User namespace is enabled (as happens in some unit files, for hardening) PID namespace must also be enabled. In that case enable the following options in the kernel configuration and recompile the kernel if necessary:

```
General setup --->
  -- Namespaces support ---> [NAMESPACES]
    # Enable or disable *both* of them:
    [ /*] User namespace [USER_NS]
    [ /*] PID Namespaces [PID_NS]
```

Contents

Installed Programs: `qtwebengine_convert_dict` and `QtWebEngineProcess` (both in `$QT6DIR/libexec`)

Installed Libraries: `libQt6Pdf.so`, `libQt6PdfQuick.so`, `libQt6PdfWidgets.so`, `libQt6WebEngineCore.so`, `libQt6WebEngineQuick.so`, `libQt6WebEngineQuickDelegatesQml.so`, and `libQt6WebEngineWidgets.so`

Installed Directories: `$QT6DIR/include/QtPdf`, `$QT6DIR/include/QtPdfQuick`, `$QT6DIR/include/QtPdfWidgets`,
`$QT6DIR/include/QtWebEngineCore`, `$QT6DIR/include/QtWebEngineQuick`,
`$QT6DIR/include/QtWebEngineWidgets`, `$QT6DIR/qml/QtWebEngine`, and
`$QT6DIR/translations/qtwebengine_locales`

Short Descriptions

<code>qtwebengine_convert_dict</code>	converts hunspell dictionaries (<code>.dic</code>) to chromium format (<code>.bdic</code>)
<code>QtWebEngineProcess</code>	is a libexec program which runs a zygote process (one that listens for spawn requests from a master process and will fork itself in response)

startup-notification-0.12

Introduction to startup-notification

The `startup-notification` package contains `startup-notification` libraries. These are useful for building a consistent manner to notify the user through the cursor that the application is loading.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/startup-notification/releases/startup-notification-0.12.tar.gz>
- Download MD5 sum: `2cd77326d4dcaed9a5a23a1232fb38e9`
- Download size: 347 KB
- Estimated disk space required: 4 MB
- Estimated build time: less than 0.1 SBU

startup-notification Dependencies

Required

[Xorg Libraries](#) and [xcb-util-0.4.1](#)

Installation of startup-notification

Install startup-notification by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -v -m644 -D doc/startup-notification.txt \
/usr/share/doc/startup-notification-0.12/startup-notification.txt
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libstartup-notification-1.so

Installed Directories: /usr/include/startup-notification-1.0 and /usr/share/doc/startup-notification-0.12

Short Descriptions

`libstartup-notification-1.so` provides the functions to assist applications in communicating with the cursor system to provide feedback to the user that the application is loading

Vulkan-Headers-1.3.294

Introduction to Vulkan-Headers

The Vulkan-Headers package contains a set of header files necessary to build and link applications against the Vulkan API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KhronosGroup/Vulkan-Headers/archive/v1.3.294/Vulkan-Headers-1.3.294.tar.gz>
- Download MD5 sum: 96778b876c8d817a98796074e9d6e6c2
- Download size: 2.2 MB
- Estimated disk space required: 102 MB
- Estimated build time: less than 0.1 SBU

Vulkan-Headers Dependencies

Required

[CMake-3.30.2](#)

Installation of Vulkan-Headers

Install Vulkan-Headers by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr -G Ninja .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/include/vk_video, /usr/include/vulkan, /usr/share/cmake/VulkanHeaders, and /usr/share/vulkan

Vulkan-Loader-1.3.294

Introduction to Vulkan-Loader

The Vulkan-Loader package contains a library which provides the Vulkan API and provides core support for graphics drivers for Vulkan.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/KhronosGroup/Vulkan-Loader/archive/v1.3.294/Vulkan-Loader-1.3.294.tar.gz>
- Download MD5 sum: 31432bec5926f78a5e302e8c0530d284
- Download size: 1.6 MB
- Estimated disk space required: 9.3 MB
- Estimated build time: less than 0.1 SBU

Vulkan-loader Dependencies

Required

[CMake-3.30.2](#), [Vulkan-Headers-1.3.294](#), [Wayland-1.23.0](#), and [Xorg Libraries](#)

Recommended (Runtime)

[Mesa-24.1.5](#) (for Vulkan drivers)

Installation of Vulkan-Loader

Note

If this package is being installed on a system where Mesa has already been installed previously, please rebuild [Mesa-24.1.5](#) after this package to install Vulkan graphics drivers.

Install Vulkan-Loader by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr   \
      -D CMAKE_BUILD_TYPE=Release    \
      -D CMAKE_SKIP_INSTALL_RPATH=ON \
      -G Ninja .. &&
ninja
```

This package does not come with a working test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

Contents

Installed Programs: None

Installed Libraries: libvulkan.so

Installed Directories: /usr/lib/cmake/VulkanLoader

Short Descriptions

libvulkan.so provides the Vulkan API and core support for graphics drivers

WebKitGTK-2.44.3

Introduction to WebKitGTK

The WebKitGTK package is a port of the portable web rendering engine WebKit to the GTK+ 3 and GTK 4 platforms.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://webkitgtk.org/releases/webkitgtk-2.44.3.tar.xz>
- Download MD5 sum: 46cf81df314acbf62f811bcfd99f4769
- Download size: 36 MB
- Estimated disk space required: 1.2 GB (159 MB installed, add 154 MB for GTK-4)
- Estimated build time: 29 SBU (for GTK-3, using parallelism=8, add 30 SBU for GTK-4)

WebKitGTK Dependencies

Required

[Cairo-1.18.0](#), [CMake-3.30.2](#), [gst-plugins-base-1.24.7](#), [gst-plugins-bad-1.24.7](#), [GTK+-3.24.43](#), [GTK-4.14.5](#), [ICU-75.1](#), [Little CMS-2.16](#), [libgudev-238](#), [libsecret-0.21.4](#), [libsoup-3.4.4](#), [libtasn1-4.19.0](#), [libwebp-1.4.0](#), [Mesa-24.1.5](#), [OpenJPEG-2.5.2](#), [Ruby-3.3.4](#), [SQLite-3.46.1](#), [unifdef-2.12](#), and [Which-2.21](#)

Recommended

[bubblewrap-0.9.0](#), [enchant-2.8.2](#), [GeoClue-2.7.1](#), [GLib-2.80.4](#) (with GObject Introspection), [hicolor-icon-theme-0.18](#), [libavif-1.1.1](#), [libjxl-0.10.3](#), [libseccomp-2.5.5](#), and [xdg-dbus-proxy-0.1.5](#)

Optional

[Gi-DocGen-2024.1](#), [harfBuzz-9.0.0](#), [Wayland-1.23.0](#), [WOFF2-1.0.2](#), [ccache](#), [Hyphen](#), [libbacktrace](#), and [libmanette](#)

Installation of WebKitGTK

Installation of WebKitGTK

Important

This package allows building with either GTK-3 or GTK-4, but not both in the same build. The GTK-4 version is needed for packages such as [Epiphany-46.3](#). Other packages, such as [Balsa-2.6.4](#) or [Evolution-3.52.4](#), require the GTK-3 version. Both versions can be installed side by side on the same system. We give build instructions for both cases below, but the only difference is the setting of `-D USE_GTK4`

First, fix a regression that occurs with some websites that use WebAssembly:

```
sed '/returnLocation.isStackArgument/,/returnLocation = canonicalLocation/d' \
-i Source/JavaScriptCore/wasm/WasmBBQJIT.cpp
```

If you want to install the GTK+-3 version of WebKitGTK, run the following commands:

```
mkdir -vp build &&
cd      build &&

cmake -D CMAKE_BUILD_TYPE=Release      \
      -D CMAKE_INSTALL_PREFIX=/usr      \
      -D CMAKE_SKIP_INSTALL_RPATH=ON   \
      -D PORT=GTK                    \
      -D LIB_INSTALL_DIR=/usr/lib     \
      -D USE_LIBBACKTRACE=OFF         \
      -D USE_LIBHYPHEN=OFF            \
      -D ENABLE_GAMEPAD=OFF           \
      -D ENABLE_MINIBROWSER=ON        \
      -D ENABLE_DOCUMENTATION=OFF    \
      -D ENABLE_WEBDRIVER=OFF         \
      -D USE_WOFF2=OFF                \
      -D USE_GTK4=OFF                 \
      -D ENABLE_BUBBLEWRAP_SANDBOX=ON \
      -W no-dev -G Ninja ..          &&
ninja
```

This package does not have a working test suite. However, there is a usable basic graphical web browser in the build directory, build/bin/MiniBrowser. If launching it fails, there is a problem with the build.

Now, as the `root` user:

```
ninja install
```

If you want to install the GTK4 version of WebKitGTK, run the following commands:

```
rm -rf * .[^.]* &&

cmake -D CMAKE_BUILD_TYPE=Release      \
      -D CMAKE_INSTALL_PREFIX=/usr      \
      -D CMAKE_SKIP_INSTALL_RPATH=ON   \
      -D PORT=GTK                    \
      -D LIB_INSTALL_DIR=/usr/lib     \
      -D USE_LIBBACKTRACE=OFF         \
      -D USE_LIBHYPHEN=OFF            \
      -D ENABLE_GAMEPAD=OFF           \
      -D ENABLE_MINIBROWSER=ON        \
      -D ENABLE_DOCUMENTATION=OFF    \
      -D USE_WOFF2=OFF                \
      -D USE_GTK4=ON                  \
      -D ENABLE_BUBBLEWRAP_SANDBOX=ON \
      -W no-dev -G Ninja ..          &&
ninja
```

Now, as the `root` user:

```
ninja install
```

As the `-D ENABLE_DOCUMENTATION=OFF` option is used, the documentation is not rebuilt nor installed now. If you need the documentation, you can either install [Gi-DocGen-2024.1](#) and remove this option from the `cmake` command, or install the pre-built documentation (only for the GTK-4 version) instead:

```
install -vdm755 /usr/share/gtk-doc/html &&
cp -rv ./Documentation/* /usr/share/gtk-doc/html
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D USE_LIBBACKTRACE=OFF`: This switch disables the use of libbacktrace for dumping a stack trace when a crash happens. You need to install [libbacktrace](#) if you wish to enable it (replacing OFF with ON or just removing the switch).

`-D USE_LIBHYPHEN=OFF`: This switch disables the default automatic hyphenation implementation. You need to install [Hyphen](#) if you wish to enable it (replacing OFF with ON or just removing the switch).

`-D ENABLE_DOCUMENTATION=OFF`: This switch disables regenerating the documentation. Remove this switch if you have [Gi-DocGen-2024.1](#) installed and wish to regenerate the documentation.

`-D ENABLE_GAMEPAD=OFF`: This switch disables gamepad support. Remove this switch if you have [libmanette](#) installed and wish to enable it.

`-D ENABLE_MINIBROWSER=ON`: This switch enables compiling and installing the MiniBrowser.

`-D ENABLE_WEBDRIVER=OFF`: This switch disables compiling and installing [WebKitWebDriver](#). We use it for the GTK-3 build because the [WebKitWebDriver](#) from the GTK-3 build will be overwritten by the GTK-4 build anyway. Remove this option if you don't plan to build this package with GTK-4.

`-D ENABLE_BUBBLEWRAP_SANDBOX=ON`: This switch enables the Bubblewrap sandbox, which helps mitigate the impact of most security vulnerabilities in this package. Change this switch to OFF if you do not want to install [bubblewrap-0.9.0](#), [libseccomp-2.5.5](#), or [xdg-dbus-proxy-0.1.5](#), but note that this may put you at risk.

`-D USE_SYSTEM_MALLOC=ON`: This switch enables building against the system installed malloc.

`-D ENABLE_GEOLOCATION=OFF`: Use this switch if you don't want to install [GeoClue-2.7.1](#).

`-D USE_AVIF=OFF`: Use this switch if you don't want to install [libavif-1.1.1](#). Note that you will not have support for AVIF images if you do this.

`-D USE_WOFF2=ON`: Use this switch if the optional package [WOFF2-1.0.2](#) is installed. This adds additional support for fonts.

`-D USE_JPEGXL=OFF`: Use this switch if the recommended package [libjxl-0.10.3](#) is not installed. Note that you will not have support for JPEG XL images if this option is passed.

Contents

Installed Program: `WebKitWebDriver`

Installed Libraries: `libjavascriptcoregtk-4.1.so`, `libjavascriptcoregtk-6.0.so`, `libwebkit2gtk-4.1.so`, and `libwebkitgtk-6.0.so`

Installed Directories: `/usr/include/webkitgtk-4.1`, `/usr/include/webkitgtk-6.0`, `/usr/lib{,exec}/webkit2gtk-4.1`, `/usr/lib{,exec}/webkitgtk-6.0`, `/usr/share/gtk-doc/html/{jsc-glib,webkit2gtk{-web-extension}}-4.1` (optional), and `/usr/share/gtk-doc/html/{jsc-glib,webkit2gtk{-web-process-extension}}-6.0`

Short Descriptions

<code>WebKitWebDriver</code>	allows debugging and automation of web pages and browsers
<code>libjavascriptcoregtk-4.1.so</code>	contains core JavaScript API functions used by <code>jsc</code> and <code>libwebkit2gtk-4.1</code>
<code>libjavascriptcoregtk-6.0.so</code>	contains core JavaScript API functions used by <code>jsc</code> and <code>libwebkitgtk-6.0</code>
<code>libwebkit2gtk-4.1.so</code>	contains the WebKit2 API functions
<code>libwebkitgtk-6.0.so</code>	contains the WebKit API functions for GTK-4 applications

xdg-desktop-portal-1.18.2

Introduction to xdg-desktop-portal

`xdg-desktop-portal` is a D-Bus service that allows applications to interact with the desktop in a safe way. Several aspects of desktop interaction, like file chooser, desktop style, etc are implemented in different D-Bus APIs, known as *portals*. Sandboxed applications benefit the most from this service since they don't need special permissions to use the portal APIs, but any application can use it. `xdg-desktop-portal` safeguards many resources and features with a user-controlled permission system. This service needs a backend implementing desktop-specific portal interfaces.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/flatpak/xdg-desktop-portal/releases/download/1.18.2/xdg-desktop-portal-1.18.2.tar.xz>
- Download MD5 sum: 20e0b28c1528e57b13cc236ba7840e03
- Download size: 683 KB
- Estimated disk space required: 51 MB (with tests)
- Estimated build time: 0.1 SBU (add 0.4 SBU for tests; with parallelism=4)

xdg-desktop-portal Dependencies

Required

[Fuse-3.16.2](#), [gdk-pixbuf-2.42.12](#), [JSON-GLib-1.8.0](#), [pipewire-1.2.3](#), and [dbus-1.14.10](#) (at runtime). Furthermore, a backend is needed at runtime for this package to be of any use, either [xdg-desktop-portal-gtk-1.15.1](#) or [xdg-desktop-portal-gnome-46.2](#) or [xdg-desktop-portal-lxqt-1.0.2](#)

Recommended

[bubblewrap-0.9.0](#) and [docutils-0.21.2](#) (for building the manual pages)

Note

Although there is an option to build the package without bubblewrap, upstream developers and LFS editors alike highly recommend to not use this possibility, as it will create a large security issue.

Optional

[GeoClue-2.7.1](#) (for the “location” portal), and [pytest-8.3.2](#) with [libportal-0.7.1](#) and [dbusmock-0.32.1](#) (for running tests)

Optional (for building the documentation)

[sphinx-8.0.2](#) with [sphinxext.opengraph](#), [sphinx_copybutton](#), [furo](#), and [flatpak](#)

Installation of xdg-desktop-portal

Install xdg-desktop-portal by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

If you have installed the needed dependencies, you can test the results by running `ninja test`. Six tests are known to fail if you are not using [gnome-terminal-3.52.2](#) and are in GNOME.

Now, as the `root` user:

```
ninja install
```

Configuring xdg-desktop-portal

Config Files

The main configuration files are `~/.config/xdg-desktop-portal/portals.conf`, `/etc/xdg-desktop-portal/portals.conf`, and `/usr/share/xdg-desktop-portal/portals.conf`. Several other locations can be searched for configuration files. See [portals.conf\(5\)](#).

Configuration Information

The various configuration files are used to choose the backend depending of various conditions. See [portals.conf\(5\)](#) for details.

Contents

Installed Programs: several daemons in /usr/libexec

Installed Librar(y,ies):None

Installed Director(y,ies):None

xdg-desktop-portal-gtk-1.15.1

Introduction to xdg-desktop-portal-gtk

`xdg-desktop-portal-gtk` is a backend for `xdg-desktop-portal`, that is using GTK and various pieces of GNOME infrastructure. This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/flatpak/xdg-desktop-portal-gtk/releases/download/1.15.1/xdg-desktop-portal-gtk-1.15.1.tar.xz>
- Download MD5 sum: 9c7836b1fe09bc914ea4c06b9c58231f
- Download size: 89 KB
- Estimated disk space required: 9.7 MB
- Estimated build time: less than 0.1 SBU

xdg-desktop-portal-gtk Dependencies

Required

[GTK+ 3.24.43](#) and [xdg-desktop-portal-1.18.2](#)

Recommended

[gnome-desktop-44.1](#) (for compiling more portal interfaces)

Installation of `xdg-desktop-portal-gtk`

Install `xdg-desktop-portal-gtk` by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Program: one daemon in /usr/libexec

Installed Library: None

Installed Directory: /usr/share/xdg-desktop-portal (if no other `xdg-desktop-portal` backend is installed)

Chapter 26. Display Managers

Display Managers are graphical programs used for starting the graphical display (currently, the X server) and providing a login capability for a Window Manager or Desktop Environment.

There are many Display Managers available. Some of the more well known include: gdm, kdm (deprecated), LightDM, lxdm, Slim, and sddm.

Among the Desktop Environments available for Linux you find: Enlightenment, GNOME, Plasma, Ixde, LXQt, and xfce.

Choosing a Display Manager or Desktop Environment is highly subjective. The choice depends on the look and feel of the packages, the resources (memory and disk space) required, and the utilities included.

In this chapter, the installation instructions of some Display Managers are presented. Later in the book, you will find other ones, which are provided as part of some Desktop Environments.

GDM-46.2

Introduction to GDM

GDM is a system service that is responsible for providing graphical logins and managing local and remote displays.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gdm/46/gdm-46.2.tar.xz>
- Download MD5 sum: 1d92ec4dd0643907fa135056aee18066
- Download size: 912 KB
- Estimated disk space required: 37 MB
- Estimated build time: 0.2 SBU

GDM Dependencies

Required

[AccountsService-23.13.9](#), [DConf-0.40.0](#), [libcanberra-0.30](#) (built after [GTK+-3.24.43](#)), and [Linux-PAM-1.6.1](#)

Optional

[keyutils-1.6.3](#)

Runtime Dependencies

[gnome-session-46.0](#), [gnome-shell-46.4](#), and [Systemd-256.4](#)

Installation of GDM

It is recommended to have a dedicated user and group to take control of the `gdm` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 21 gdm &&
useradd -c "GDM Daemon Owner" -d /var/lib/gdm -u 21 \
        -g gdm -s /bin/false gdm &&
passwd -ql gdm
```

Install GDM by running the following commands:

```
mkdir build &&
cd      build &&

meson setup .. \
        --prefix=/usr \
        --buildtype=release \
        -D gdm-xsession=true \
        -D run-dir=/run/gdm &&
ninja
```

This package does not come with a usable test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D initial-vt=7`: Use this switch to make GDM start on VT7 instead of the first free VT.

`-D default-pam-config=lfs`: Use this switch if you did not create the `/etc/lfs-release` file or distribution auto detection will fail and you will be unable to use GDM.

`-D gdm-xsession=true`: This enables the installation of the GDM Xsession file.

Configuring GDM

Config Files

/etc/gdm/custom.conf

Configuration Information

The GDM daemon is configured using the `/etc/gdm/custom.conf` file. Default values are stored in GSettings in the `gdm.schemas` file. It is recommended that end-users modify the `/etc/gdm/custom.conf` file because the schemas file may be overwritten when the user updates their system to have a newer version of GDM.

On some systems with NVIDIA GPUs, GDM will hide Wayland sessions by default. This is often done to prevent users from encountering problems with buggy drivers, which can result in system lockups, application crashes, power management problems, and graphics slowdowns. If you have an NVIDIA GPU and still want to try running Wayland sessions anyway, execute the following command as the `root` user:

```
ln -s /dev/null /etc/udev/rules.d/61-gdm.rules
```

Systemd Unit

To start the `gdm` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable gdm
```

Configuring GDM Auto-Suspend

GDM will suspend the system when the greeter screen has been running for a while without any interactive input. If you want to disable auto-suspending for any reason (for example if the system is hosting some services besides functioning as a desktop system), as the `root` user, issue:

```
su gdm -s /bin/bash
      -c "dbus-run-session
          gsettings set org.gnome.settings-daemon.plugins.power \
              sleep-inactive-ac-type \
              nothing"
```

Contents

Installed Programs: `gdm`, `gdmflexiserver`, and `gdm-screenshot`

Installed Libraries: `libgdm.so` and `pam_gdm.so` (PAM Module)

Installed Directories: `/etc/gdm`, `/usr/include/gdm`, `/usr/share/gdm`, and `/usr/share/help/*/gdm`

Short Descriptions

`gdm` is a login prompt for GNOME
`gdm-screenshot` is a screenshot tool for GDM

lightdm-1.32.0

Introduction to Lightdm

The lightdm package contains a lightweight display manager based upon GTK.

This package is known to build and work properly using an LFS 12.2 platform.

Lightdm Package Information

- Download (HTTP): <https://github.com/CanonicalLtd/lightdm/releases/download/1.32.0/lightdm-1.32.0.tar.xz>
- Download MD5 sum: e62a5da6c35f612e4d9575eda5c8d467
- Download size: 508 KB
- Estimated disk space required: 20 MB
- Estimated build time: 0.2 SBU

Additional download

Note

The greeter is a program to present a graphical login screen. There are several alternative greeters, but the gtk+ package is the reference implementation. For a list of other greeters, see <https://en.wikipedia.org/wiki/LightDM>.

- Download (HTTP): <https://github.com/Xubuntu/lightdm-gtk-greeter/releases/download/lightdm-gtk-greeter-2.0.9/lightdm-gtk-greeter-2.0.9.tar.gz>
- Download MD5 sum: 35752d730f39293c2bbe884ef6963830
- Download size: 592 KB
- Estimated disk space required: 5.2 MB
- Estimated build time: less than 0.1 SBU

Lightdm Dependencies

Required

[Exo-4.18.0](#) (for the greeter), [libgcrypt-1.11.0](#), [itstool-2.0.7](#), [Linux-PAM-1.6.1](#), and [Xorg-Server-21.1.13](#) (Runtime)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [libxklavier-5.4](#), and [Vala-0.56.17](#)

Optional

[AccountsService-23.13.9](#) (run time), [at-spi2-core-2.52.0](#), [GTK-Doc-1.34.0](#), [qt5-components-5.15.14](#), [libido](#), and [libindicator](#)

Installation of Lightdm

First, create a dedicated user and group to take control of the `lightdm` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 65 lightdm      &&
useradd -c "Lightdm Daemon" \
         -d /var/lib/lightdm \
         -u 65 -g lightdm \
         -s /bin/false lightdm
```

Install lightdm by running the following commands:

```
./configure --prefix=/usr \
            --libexecdir=/usr/lib/lightdm \
            --localstatedir=/var \
            --bindir=/usr/bin \
            --sysconfdir=/etc \
            --disable-static \
            --disable-tests \
            --with-greeter-user=lightdm \
            --with-greeter-session=lightdm-gtk-greeter \
            --docdir=/usr/share/doc/lightdm-1.32.0 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install                  &&
cp tests/src/lightdm-session /usr/bin          &&
sed -i '1 s/sh/bash --login/' /usr/bin/lightdm-session &&
rm -rf /etc/init                &&
install -v -dm755 -o lightdm -g lightdm /var/lib/lightdm &&
install -v -dm755 -o lightdm -g lightdm /var/lib/lightdm-data &&
install -v -dm755 -o lightdm -g lightdm /var/cache/lightdm   &&
install -v -dm770 -o lightdm -g lightdm /var/log/lightdm
```

Now build the greeter:

```
tar -xf ../lightdm-gtk-greeter-2.0.9.tar.gz &&
cd lightdm-gtk-greeter-2.0.9 &&
```

```

./configure --prefix=/usr \
--libexecdir=/usr/lib/lightdm \
--sbindir=/usr/bin \
--sysconfdir=/etc \
--with-libxklavier \
--enable-kill-on-sigterm \
--disable-libido \
--disable-libindicator \
--disable-static \
--disable-maintainer-mode \
--docdir=/usr/share/doc/lightdm-gtk-greeter-2.0.9 &&
make

```

Now, as the `root` user:

```
make install
```

Note

If you installed Xorg in /opt, you will need to create a symbolic link so lightdm can find the Xorg server. As the `root` user:

```
ln -sf /opt/xorg/bin/Xorg /usr/bin/X
```

Command Explanations

`sed ... /usr/bin/lightdm-session`: This command ensures that the initial login via the greeter sources /etc/profile and `~/.bash_profile`. Without this, commands that depend on different environment variables may not work as expected.

Configuring lightdm

Config Files

`/etc/lightdm/{lightdm,users,keys,lightdm-gtk-greeter}.conf`

The configuration files offer many options. If, for instance, you have a multiple monitor setup but prefer the login window to be displayed on only one monitor, set 'active-monitor=<monitor-name>' in `lightdm-gtk-greeter.conf`. For example you could use `active-monitor=HDMI-1`. You can determine the monitor-names with `xrandr --listmonitors`. Note that the monitor names may change if you change the graphics driver and you may have to adjust the configuration accordingly.

Systemd Unit

Install the `lightdm.service` unit included in the [blfs-systemd-units-20240801](#) package:

```
make install-lightdm
```

Available Sessions

The greeter offers a list of available sessions, depending on the Window Managers and Desktop Environments installed. The list includes sessions which have a corresponding `.desktop` file installed under `/usr/share/xsessions`. Most of the Window Managers and Desktop Environments automatically provide those files, but if necessary, you may include a custom one.

Contents

Installed Programs: `dm-tool`, `lightdm`, and `lightdm-gtk-greeter`

Installed Libraries: `liblightdm-gobject-1.so`

Installed Directories: `/etc/lightdm`, `/etc/apparmor.d`, `/usr/lib/lightdm`, `/usr/include/lightdm-gobject-1`,
`/usr/share/help/C/lightdm` `/usr/share/gtk-doc/html/lightdm-gobject-1`, `/usr/share/doc/lightdm-gtk-greeter-2.0.9`, `/var/lib/lightdm`, `/var/lib/lightdm-data`, `/var/cache/lightdm`, and `/var/log/lightdm`

Short Descriptions

`lightdm` is a display and login manager

`lightdm-gtk-greeter` is an auxiliary process that displays the greeter, a graphical user interface that performs user authentication and initiates the selected window manager or display environment

sddm-0.21.0

Introduction to sddm

The sddm package contains a lightweight display manager based upon Qt and QML.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/sddm/sddm/archive/v0.21.0/sddm-0.21.0.tar.gz>
- Download MD5 sum: e32a35c282d9be3360737eefbe25b5fa
- Download size: 3.4 MB
- Estimated disk space required: 24 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

SDDM Dependencies

Required

[CMake-3.30.2](#), [extra-cmake-modules-6.5.0](#), and [Qt-6.7.2](#)

Recommended

[docutils-0.21.2](#) (for the man pages), [Linux-PAM-1.6.1](#), and [UPower-1.90.4](#)

Installation of SDDM

First, create a dedicated user and group to take control of the `sddm` daemon after it is started. Issue the following commands as the `root` user:

```
groupadd -g 64 sddm &&
useradd -c "sddm Daemon" \
-d /var/lib/sddm \
-u 64 -g sddm \
-s /bin/false sddm
```

Install sddm by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D RUNTIME_DIR=/run/sddm \
-D BUILD_MAN_PAGES=ON \
-D BUILD_WITH_QT6=ON \
-D DATA_INSTALL_DIR=/usr/share/sddm \
-D DBUS_CONFIG_FILENAME=sddm_org.freedesktop.DisplayManager.conf \
... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -v -dm755 -o sddm -g sddm /var/lib/sddm
/usr/bin/sddm --example-config > /etc/sddm.conf
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply additional compiler optimizations.

`-D DBUS_CONFIG_FILENAME=sddm_org.freedesktop.DisplayManager.conf`: This switch prevents the file `/etc/dbus-1/system.d/org.freedesktop.DisplayManager.conf` from being overwritten, as it may be used by other DM's.

`-D BUILD_MAN_PAGES=ON`: This switch is used to build and install man pages.

Configuring SDDM

Config Files

`/etc/sddm.config`

Normally, you want to edit this file. For example, if Xorg is installed in `/opt`, use your preferred editor as the `root` user to replace the default `XauthPath` value by `/opt/xorg/bin/xauth`. Or, as the `root` user, issue:

```
sed -i.orig '/ServerPath/ s|usr|opt|xorg|' /etc/sddm.conf
```

This command will do the substitution and create a copy of the original file with name `/etc/sddm.conf.orig`.

From now on, we will describe how to modify configurations using sed. Of course, you may instead use your preferred editor as the `root` user.

For security reasons, you normally want the default `ServerArguments==nolisten tcp`, unless a remote machine needs access to the local X server. In that case, as the `root` user, issue:

```
sed -i 's/-nolisten tcp//' /etc/sddm.conf
```

Desktop (Notebook) users, normally want the Num Lock key on (off). For that, as `root`, issue:

```
sed -i '/Numlock/s/none/on/' /etc/sddm.conf
```

for Desktop users. For Notebook users, replace `/on/` by `/off/` in the command above.

By default, a virtual keyboard is presented for the user. If this is not desired, run as `root`:

```
sed -i 's/qtvirtualkeyboard//' /etc/sddm.conf
```

Boot Script

Enable the pre-installed systemd unit by running the following command as the `root` user:

```
systemctl enable sddm
```

Linux PAM Configuration

Note

The install procedure above installed a set of PAM configuration files. These procedures overwrite them and use versions compatible with a BLFS environment.

If you have built sddm with Linux PAM support, create the necessary configuration files by running the following commands as the `root` user:

```
cat > /etc/pam.d/sddm << "EOF" &&
# Begin /etc/pam.d/sddm

auth      requisite      pam_nologin.so
auth      required       pam_env.so

auth      required       pam_succeed_if.so uid >= 1000 quiet
auth      include        system-auth

account  include        system-account
password include       system-password

session  required       pam_limits.so
session  include        system-session
```

```

# End /etc/pam.d/sddm
EOF

cat > /etc/pam.d/sddm-autologin << "EOF" &&
# Begin /etc/pam.d/sddm-autologin

auth    requisite      pam_nologin.so
auth    required       pam_env.so

auth    required      pam_succeed_if.so uid >= 1000 quiet
auth    required      pam_permit.so

account  include      system-account

password required    pam_deny.so

session  required     pam_limits.so
session  include      system-session

# End /etc/pam.d/sddm-autologin
EOF

cat > /etc/pam.d/sddm-greeter << "EOF"
# Begin /etc/pam.d/sddm-greeter

auth    required      pam_env.so
auth    required      pam_permit.so

account  required     pam_permit.so
password required    pam_deny.so
session  required     pam_unix.so
-session optional    pam_systemd.so

# End /etc/pam.d/sddm-greeter
EOF

```

Available Sessions

The greeter offers a list of available sessions, depending on the Window Managers and Desktop Environments installed. The list includes sessions which have a corresponding .desktop file installed under `/usr/share/xsessions` or `/usr/share/wayland-sessions`. Most of the Window Managers and Desktop Environments automatically provide those files, but if necessary, you may include a custom one.

Themes

Three themes are installed at `/usr/share/sddm/themes`: elarun, maldives, and maya. There is also a default theme, which is not present in that directory. You can install other themes in that directory. In order to change the theme, you need to edit `/etc/sddm.conf`, to change the default (empty) theme, replacing `Current=` with `Current=<new theme>`, e.g. `Current=maldives`.

In order to see the theme without leaving the session, issue:

```
sddm-greeter --test-mode --theme <theme path>
```

Contents

Installed Programs: sddm and sddm-greeter

Installed Libraries: None

Installed Directories: \$QT5DIR/qml/SddmComponents, `/usr/share/sddm`, and `/var/lib/sddm`

Short Descriptions

<code>sddm</code>	is a display and login manager based on Qt libraries.
<code>sddm-greeter</code>	is an auxiliary process that displays the greeter, a graphical user interface that performs user authentication and initiates the selected window manager or display environment.

Chapter 27. Window Managers

Window Managers and Desktop Environments are the primary user interfaces into the X Window System. A window manager is a program that controls the appearance of windows and provides the means by which the user can interact with them. A

Desktop Environment provides a more complete interface to the operating system, and provides a range of integrated utilities and applications.

There are many Window Managers available. Some of the more well known ones include fvwm2, Window Maker, AfterStep, Enlightenment, Sawfish, and Blackbox.

The Desktop Environments available for Linux are GNOME, KDE, and XFce.

Choosing a Window Manager or Desktop Environment is highly subjective. The choice depends on the look and feel of the packages, the resources (RAM, disk space) required, and the utilities included. One web site that provides a very good summary of what is available, screenshots, and their respective features is [Window Managers for X](#).

In this chapter, the installation instructions of several Window Managers and one lightweight Desktop Environment are presented. Later in the book, both KDE and GNOME have their own sections.

Fluxbox-1.3.7

Introduction to Fluxbox

The Fluxbox package contains a window manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/fluxbox/fluxbox-1.3.7.tar.xz>
- Download MD5 sum: b44afdf0ee1e64624c23115aa51dcd55
- Download size: 772 KB
- Estimated disk space required: 101 MB
- Estimated build time: 0.9 SBU

Fluxbox Dependencies

Required

[a graphical environment](#)

Optional

[dbus-1.14.10](#) (runtime), [FriBidi-1.0.15](#), and [imlib2-1.12.3](#) (if you wish to use other image formats in addition to XPM)

Installation of Fluxbox

First, fix a build failure when building with gcc-11.1:

```
sed -i '/text_prop.value > 0/s/>/!=/' util/fluxbox-remote.cc
```

Install Fluxbox by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not have a working test suite.

Now, as the `root` user:

```
make install
```

Configuring Fluxbox

Config Files

`~/.fluxbox/init`, `~/.fluxbox/keys`, and `~/.fluxbox/menu`

Configuration Information

If Fluxbox is the only Window Manager you want to use, you can start it with an `.xinitrc` file in your home folder. Be sure to backup your current `.xinitrc` before proceeding.

```
echo startfluxbox > ~/.xinitrc
```

Or alternatively, if you use a login manager like [GDM-46.2](#) or [lightdm-1.32.0](#), and would like to be able to choose Fluxbox at the login prompt, create a `fluxbox.desktop` file. As `root`:

```
mkdir -pv /usr/share/xsessions &&
cat > /usr/share/xsessions/fluxbox.desktop << "EOF"
[Desktop Entry]
Encoding=UTF-8
Name=Fluxbox
Comment=This session logs you into Fluxbox
Exec=startfluxbox
Type=Application
EOF
```

If you didn't install [GDM-46.2](#) or [lightdm-1.32.0](#) in `/usr`, then change that command to fit the prefix you chose.

Now create the Fluxbox configuration files:

```
mkdir -v ~/.fluxbox &&
cp -v /usr/share/fluxbox/init ~/.fluxbox/init &&
cp -v /usr/share/fluxbox/keys ~/.fluxbox/keys
```

To generate the application menu, first you may wish to run `fluxbox-generate_menu -h`, in order to choose any `<user_options>`, then issue:

```
cd ~/.fluxbox &&
fluxbox-generate_menu <user_options>
```

Alternately, copy a pregenerated menu:

```
cp -v /usr/share/fluxbox/menu ~/.fluxbox/menu
```

Menu items are added by editing `~/.fluxbox/menu`. The syntax is explained on the `fluxbox` man page.

If you want to use an image as your desktop background, copy the theme you like into `~/.fluxbox`. Then add a line to make it use the correct image. In the following command, change `<theme>` for the name of the theme you want and change `</path/to/nice/image.ext>` to point to the actual image you want to use, where `ext` must be `xpm`, if [imlib2-1.12.3](#) is not installed to allow other image formats.

```
cp -r /usr/share/fluxbox/styles/<theme> ~/.fluxbox/theme &&
sed -i 's,(session.styleFile:).*,\1 ~/.fluxbox/theme,' ~/.fluxbox/init &&
[ -f ~/.fluxbox/theme ] &&
echo "background.pixmap: </path/to/nice/image.ext>" >> ~/.fluxbox/theme ||
[ -d ~/.fluxbox/theme ] &&
echo "background.pixmap: </path/to/nice/image.ext>" >> ~/.fluxbox/theme/theme.cfg
```

In some locales the font specified in the theme may not contain the needed characters. This results in menus with blank items. You can fix this by editing `~/.fluxbox/theme` with a text editor and altering it so that it names a suitable font.

Contents

- Installed Programs:** fluxbox, fbsetbg, fbsetroot, fluxbox-generate_menu, startfluxbox, fbrun, fluxbox-remote, and fluxbox-update_configs
- Installed Libraries:** None
- Installed Directories:** /usr/share/fluxbox and `~/.fluxbox`

Short Descriptions

<code>fluxbox</code>	is a window manager for X11 based on Blackbox 0.61.0
<code>fbsetbg</code>	is a utility that sets the background image. It requires one of the following at runtime: <code>Esetroot</code> , <code>wmsetbg</code> , <code>feh</code> , <code>hsetroot</code> , <code>chbg</code> , <code>display</code> , <code>qiv</code> , <code>xv</code> , <code>xsri</code> , <code>xli</code> , OR <code>xsetbg</code> . It also requires <code>which</code> if <code>Esetroot</code> is found
<code>fbsetroot</code>	is a utility to change root window appearance based on the Blackbox application bsetroot

<code>fluxbox-generate_menu</code>	is a utility that generates a menu by scanning your <code>PATH</code>
<code>startfluxbox</code>	is a session startup script that allows for command executions prior to <code>fluxbox</code> starting
<code>fbrun</code>	displays a run dialog window
<code>fluxbox-remote</code>	provides command line access to key commands for Fluxbox
<code>fluxbox-update_configs</code>	use to manage config files (reload, update, test)

IceWM-3.6.0

Introduction to IceWM

IceWM is a window manager with the goals of speed, simplicity, and not getting in the user's way.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/ice-wm/icewm/archive/3.6.0/icewm-3.6.0.tar.gz>
- Download MD5 sum: a94dde82caae27f01d57fba7ba12225a
- Download size: 2.4 MB
- Estimated disk space required: 47 MB
- Estimated build time: 0.4 SBU

IceWM Dependencies

Required

[CMake-3.30.2](#), [imlib2-1.12.3](#), and [a graphical environment](#)

Optional

[asciidoc-10.2.1](#) (for regenerating the HTML documentation), [FriBidi-1.0.15](#) (for languages written right to left), [librsvg-2.58.3](#), [libao-1.2.0](#), [libsndfile-1.2.2](#), [alsa-lib-1.2.12](#) (for the experimental icesound program), and [gdk-pixbuf-xlib](#)

Installation of IceWM

Install IceWM by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D CFGDIR=/etc \
-D ENABLE_LTO=ON \
-D DOCDIR=/usr/share/doc/icewm-3.6.0 \
... &&
make
```

This package does not have a working test suite.

Now, as the `root` user:

```
make install
```

This package creates two `.desktop` files in the `/usr/share/xsessions/` directory. Both are not needed in a BLFS system, so prevent the extra file showing up as an option in a display manager. As the `root` user:

```
rm -v /usr/share/xsessions/icewm.desktop
```

Command Explanations

`-D ENABLE_LTO=ON`: This option enables Link Time Optimization and is required for this package.

Configuring IceWM

Config Files

~/.icewm/keys, ~/.icewm/menu, and ~/.icewm/preferences, and ~/.icewm/toolbar, and ~/.icewm/winoptions. The default versions are installed in /usr/share/icewm/ and will be used if you have not copied them to ~/.icewm.

Configuration Information

If IceWM is the only Window Manager you want to use, you can start it with an .xinitrc file in your home folder. Be sure to backup your current .xinitrc before proceeding.

```
echo icewm-session > ~/.xinitrc
```

Now create the IceWM configuration files:

```
mkdir -pv ~/.icewm
cp -v /usr/share/icewm/keys ~/.icewm/keys
cp -v /usr/share/icewm/menu ~/.icewm/menu
cp -v /usr/share/icewm/preferences ~/.icewm/preferences &&
cp -v /usr/share/icewm/toolbar ~/.icewm/toolbar
cp -v /usr/share/icewm/winoptions ~/.icewm/winoptions
```

You can now edit these files to meet your requirements. In particular, review the preferences file. You can use Logout -> Restart-IceWM on the main menu to load your changed preferences, but changes to the background only take effect when IceWM is started.

At this point you can either modify the traditional menu files to suit your requirements, or use the newer icewm-menu-fdo described later.

The syntax of the menus is explained in the help files, which you can access by running help from the menu, but some of the detail is out of date and the default selections in the menus (a few old applications on the main menu, everything else on the Programs menu) will benefit from being updated to meet your needs. The following examples are provided to encourage you to think about how you wish to organise your menus. Please note the following:

- If a program listed in the menu has not been installed, it will not appear when the menu is displayed. Similarly, if the program exists but the specified icon does not, no icon will be displayed in the menu.
- The icons can be either .xpm or .png files, and there is no need to specify the extension. If the icon is located in the "library" (/usr/share/icewm/icons) there is no need to specify the path.
- Most programs are in sub-menus, and the main menu will always append entries for windows, help, settings, logout at the bottom.
- An icon for firefox was copied to the library directory and given a meaningful name. The icon for xine is xine.xpm which was installed to a pixmap directory.
- The default toolbar is not altered.

If you wish to use this traditional method, there are more examples in previous releases of this book (e.g. BLFS-7.8).

Alternatively, you can create a menu which conforms to the FDO Desktop Menu Specifications, where programs can be found because they have a .desktop file in the XDG_DATA_HOME or XDG_DATA_DIR directories. Unlike most windowmanagers, icewm does not search for programs when the menu is invoked, so if you take this route you will need to rerun the following command after installing or removing programs:

```
icewm-menu-fdo >~/icewm/menu
```

If you wish to put icons on your desktop, you will need to install a program such as [rox-filer](#) which provides a pinboard. If you do that you will no longer be able to access the menu by right-clicking on the desktop, you will have to use the IceWM button. To ensure that the rox pinboard is running, the following commands will put it in the startup file:

```
cat > ~/icewm/startup << "EOF"
rox -p Default &
EOF &&
chmod +x ~/icewm/startup
```

Tip

There are a number of keyboard shortcuts in IceWM:

- Ctrl + Alt + F_N : go to ttyN.
- Ctrl + Alt + N : go to desktop number N
- Ctrl + Alt + Space : open a box on the taskbar where you can key in the name of an application and run it.

Contents

Installed Programs: icehelp, icesh, icesound, icewm, icewm-menu-fdo, icewm-menu-xrandr, icewm-session, icewm-set-gnomewm, icewmbg, and icewmhint

Installed Libraries: None

Installed Directories: /usr/share/doc/icewm-3.6.0, /usr/share/icewm and ~./icewm

Short Descriptions

icehelp	is used to display the html manual
icesh	is a command-line window manager which can be used in ~/.icewm/startup
icesound	plays audio files on GUI events raised by IceWM
icewm	is the window manager
icewm-menu-fdo	can create a file in a format suitable for an IceWM menu, which lists those programs currently installed in a layout conforming to the FDO Desktop Menu Specifications
icewm-menu-xrandr	is a helper program used to manage multi-screen configurations
icewm-session	runs icewmbg, icewm, icewmtray, startup, shutdown (i.e. startup and shutdown scripts are run if installed)
icewm-set-gnomewm	is a script to set the GNOME window manager to icewm using gconftool
icewmbg	is used to set the background, according to the various <code>DesktopBackground</code> settings in the preferences
icewmhint	is used internally to provide hints to the user

openbox-3.6.1

Introduction to openbox

Openbox is a highly configurable desktop window manager with extensive standards support. It allows you to control almost every aspect of how you interact with your desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://openbox.org/dist/openbox/openbox-3.6.1.tar.gz>
- Download MD5 sum: b72794996c6a3ad94634727b95f9d204
- Download size: 944 KB
- Estimated disk space required: 21 MB
- Estimated build time: 0.3 SBU

Openbox Dependencies

Required

[a graphical environment](#) and [Pango-1.54.0](#) (compiled with support for libXft)

Optional

[dbus-1.14.10](#) (runtime), [imlib2-1.12.3](#) (to enable icons in the right click menu), [ImageMagick-7.1.1-36](#) (to show desktop backgrounds as seen in the Configuration Information section below), [PyXDG-0.28](#), [startup-notification-0.12](#), and [librsvg-2.58.3](#)

Installation of Openbox

Note

If XORG_PREFIX is not `/usr`, tell `gcc` about it:

```
export LIBRARY_PATH=$XORG_PREFIX/lib
```

If you only installed the Python 3 PyXDG module convert one of the scripts to Python 3:

```
2to3-3.12 -w data/autostart/openbox-xdg-autostart &&
sed 's/python/python3/' -i data/autostart/openbox-xdg-autostart
```

Install Openbox by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --disable-static \
            --docdir=/usr/share/doc/openbox-3.6.1 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

This package creates three `.desktop` files in the `/usr/share/xsessions/` directory. Two of these are not appropriate in a BLFS system, so prevent the extra files showing up as options in a display manager. As the `root` user:

```
rm -v /usr/share/xsessions/openbox-{gnome,kde}.desktop
```

Command Explanations

`--sysconfdir=/etc`: This option puts Openbox's configuration files in `/etc/xdg/openbox` instead of `/usr/etc/xdg/openbox`.

`--docdir=/usr/share/doc/openbox-3.6.1`: this puts a few files in a versioned directory in `/usr/share/doc`.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Configuring Openbox

Config Files

`/etc/xdg/openbox/autostart`, `/etc/xdg/openbox/menu.xml`, `/etc/xdg/openbox/rc.xml`, `~/.config/openbox/autostart`,
`~/.config/openbox/menu.xml` and `~/.config/openbox/rc.xml`

Configuration Information

Openbox's right click menu can be used to launch programs. The menu itself is configured with 2 files, `/etc/xdg/openbox/menu.xml` and `~/.config/openbox/menu.xml`. To make changes to the menu, copy `/etc/xdg/openbox/menu.xml` to `~/.config/openbox/menu.xml` and edit it:

```
cp -rf /etc/xdg/openbox ~/.config
```

To have icons in your right click menu requires installing [imlib2-1.12.3](#) before you install Openbox. To set an icon for an entry in the menu edit `~/.config/openbox/menu.xml` and add an icon to the `<item>` tag like this:

```
<item label="Mplayer" icon="/usr/share/pixmaps/mplayer.png">
```

Many other aspects of Openbox's behavior are configured with `~/.config/openbox/rc.xml` such as what keybindings are used to launch programs or which mouse button launches the main menu.

Details of the theme that Openbox applies to window decorations are configured in `~/.config/openbox/rc.xml`. You can get a list of the available themes with the command:

```
ls -d /usr/share/themes/*openbox-3 | sed 's#.*es/##;s#/o.*##'
```

Starting Openbox

To automatically start `openbox` when you start Xorg:

```
echo openbox > ~/.xinitrc
```

If you want to set a background image to your desktop you can use [display](#) and launch it from `~/.xinitrc` just before `openbox`:

```
cat > ~/.xinitrc << "EOF"
display -backdrop -window root /path/to/beautiful/picture.jpeg
exec openbox
EOF
```

Or if you like a bit of variety, put a selection of images in a folder (in this example, the directory `~/.config/backgrounds`) and choose one at random each time you `xinit`:

```
cat > ~/.xinitrc << "EOF"
# make an array which lists the pictures:
picture_list=(~/config/backgrounds/*)
# create a random integer between 0 and the number of pictures:
random_number=$(( ${RANDOM} % ${#picture_list[@]} ))
# display the chosen picture:
display -backdrop -window root "${picture_list[$random_number]}"
exec openbox
EOF
```

If you like to have the numlock key set when you start Xorg, install [Numlockx](#) and add that to your `xinitrc`. Another useful application is [dbus-1.14.10](#):

```
cat > ~/.xinitrc << "EOF"
. /etc/profile
picture_list=(~/config/backgrounds/*)
random_number=$(( ${RANDOM} % ${#picture_list[*]} ))
display -backdrop -window root "${picture_list[$random_number]}"
numlockx
eval $(dbus-launch --auto-syntax --exit-with-session)
lxpanel &
exec openbox
EOF
```

Contents

Installed Programs: `gdm-control`, `gnome-panel-control`, `obxprop`, `openbox`, `openbox-gnome-session`, `openbox-kde-session` and `openbox-session`

Installed Libraries: `libobrender.so` and `libobt.so`

Installed Directories: `/etc/xdg/openbox`, `/usr/include/openbox`, `/usr/share/doc/openbox-3.6.1` and `/usr/share/themes`.

Short Descriptions

<code>gdm-control</code>	is a command line tool to send signals to GDM
<code>gnome-panel-control</code>	is a command line utility to invoke the Gnome Panel run dialog/menu
<code>obxprop</code>	is a tool for displaying the properties on an x window. It has a similar functionality to <code>xprop</code> , but allows you to see UTF-8 strings as text
<code>openbox</code>	is a standards compliant, highly configurable, window manager
<code>openbox-gnome-session</code>	is a script to launch an Gnome session with Openbox as your window manager from your <code>~/.xinitrc</code>
<code>openbox-kde-session</code>	is a script to launch an KDE session with Openbox as your window manager from your <code>~/.xinitrc</code>
<code>openbox-session</code>	is a script to launch an Openbox session from your <code>~/.xinitrc</code>
<code>libobrender.so</code>	contains the functions used by Openbox for theme rendering
<code>libobt.so</code>	is the Openbox toolkit library

Window Managers and Desktop Environments can use icons from different sources. Generally icons are installed in `/usr/share/icons` and are independent of distribution.

adwaita-icon-theme-46.2

Introduction to Adwaita Icon Theme

The Adwaita Icon Theme package contains an icon theme for GTK+ 3 and GTK 4 applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/adwaita-icon-theme/46/adwaita-icon-theme-46.2.tar.xz>
- Download MD5 sum: 6829461ddc1dd0a9b7d115ccc0c186ac
- Download size: 4.4 MB
- Estimated disk space required: 30 MB
- Estimated build time: less than 0.1 SBU

Adwaita Icon Theme Dependencies

Required

[GTK+-3.24.43](#) or [GTK-4.14.5](#), and [librsvg-2.58.3](#)

Optional

[git-2.46.0](#), [Inkscape-1.3.2](#), and [Icon Tools](#)

Installation of Adwaita Icon Theme

Install Adwaita Icon Theme by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr .. &&
ninja
```

This package does not come with a test suite.

Remove any old icons and install the new ones. As the `root` user:

```
rm -rf /usr/share/icons/Adwaita/ &&
ninja install
```

Contents

Installed Programs: None

Installed Library: None

Installed Directory: /usr/share/icons/Adwaita

breeze-icons-6.5.0

Introduction to Breeze Icons

The Breeze Icons package contains the default icons for KDE Plasma applications, but it can be used for other window environments.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/breeze-icons-6.5.0.tar.xz>
- Download MD5 sum: 25c0a753edc62593ecca9f56a34f7ef1
- Download size: 2.1 MB
- Estimated disk space required: 251 MB
- Estimated build time: 0.7 SBU

Breeze Icons Dependencies

Required

[extra-cmake-modules-6.5.0](#) and [Qt-6.7.2](#)

Optional

[KDE Frameworks-6.5.0](#), [libxml2-2.13.3](#), and [lxml-5.3.0](#)

Installation of Breeze Icons

Install Breeze Icons by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D BUILD_TESTING=OFF \
      -W no-dev ..
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D BUILD_TESTING=OFF`: This parameter disables building the testing framework, which requires some modules from [KDE Frameworks-6.5.0](#).

Contents

Installed Programs: None

Installed Library: None

Installed Directory: /usr/share/icons/breeze{-dark}

gnome-icon-theme-3.12.0

Introduction to GNOME Icon Theme

The GNOME Icon Theme package contains an assortment of non-scalable icons of different sizes and themes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-icon-theme/3.12/gnome-icon-theme-3.12.0.tar.xz>
- Download MD5 sum: f14bed7f804e843189ffa7021141add
- Download size: 17 MB
- Estimated disk space required: 85 MB
- Estimated build time: 0.5 SBU

GNOME Icon Theme Dependencies

Required

[GTK+-3.24.43](#), [hicolor-icon-theme-0.18](#), and [icon-naming-utils-0.8.90](#)

Installation of GNOME Icon Theme

Install GNOME Icon Theme by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: /usr/share/icons/gnome

gnome-icon-theme-extras-3.12.0

Introduction to GNOME Icon Theme Extras

The GNOME Icon Theme Extras package contains extra icons for the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-icon-theme-extras/3.12/gnome-icon-theme-extras-3.12.0.tar.xz>
- Download MD5 sum: 91f8f7e35a3d8d926716d88b8b1e9a29
- Download size: 1.7 MB
- Estimated disk space required: 12 MB
- Estimated build time: less than 0.1 SBU

GNOME Icon Theme Extras Dependencies

Required

[gnome-icon-theme-3.12.0](#)

Optional

[git-2.46.0](#) and [Inkscape-1.3.2](#)

Installation of GNOME Icon Theme Extras

Install GNOME Icon Theme Extras by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

gnome-icon-theme-symbolic-3.12.0

Introduction to GNOME Icon Theme Symbolic

The GNOME Icon Theme Symbolic package contains symbolic icons for the default GNOME icon theme.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-icon-theme-symbolic/3.12/gnome-icon-theme-symbolic-3.12.0.tar.xz>
- Download MD5 sum: 3c9c0e6b9fa04b3cbbb84da825a26fd9
- Download size: 228 KB
- Estimated disk space required: 6.8 MB
- Estimated build time: less than 0.1 SBU

GNOME Icon Theme Symbolic Dependencies

Required

[gnome-icon-theme-3.12.0](#)

Optional

[git-2.46.0](#) and [Inkscape-1.3.2](#)

Installation of GNOME Icon Theme Symbolic

Install GNOME Icon Theme Symbolic by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/icons/gnome/scalable

gnome-themes-extra-3.28

Introduction to GNOME Themes Extra

The GNOME Themes Extra package, formerly known as GNOME Themes Standard, contains various components of the default GNOME theme.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-themes-extra/3.28/gnome-themes-extra-3.28.tar.xz>
- Download MD5 sum: f9f2c6c521948da427f702372e16f826
- Download size: 2.8 MB
- Estimated disk space required: 40 MB
- Estimated build time: 0.3 SBU

GNOME Themes Extra Dependencies

Required

[GTK+-3.24.43](#) with [librsvg-2.58.3](#)

Installation of GNOME Themes Extra

Install GNOME Themes Extra by running the following commands:

```
./configure --prefix=/usr --disable-gtk2-engine &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-gtk2-engine`: This switch disables the GTK+-2 theming engine because GTK+-2 is no longer part of BLFS.

`--disable-gtk3-engine`: This switch disables the GTK+-3 theming engine.

Contents

Installed Programs: None

Installed Library: None

Installed Directories: /usr/share/icons/HighContrast, /usr/share/themes/Adwaita, /usr/share/themes/Adwaita-dark, and /usr/share/themes/HighContrast

hicolor-icon-theme-0.18

Introduction to hicolor-icon-theme

The hicolor-icon-theme package contains a default fallback theme for implementations of the icon theme specification.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://icon-theme.freedesktop.org/releases/hicolor-icon-theme-0.18.tar.xz>
- Download MD5 sum: ef14f3af03bcde9ed134aad626bdbaad
- Download size: 32 KB
- Estimated disk space required: 644 KB
- Estimated build time: less than 0.1 SBU

Installation of hicolor-icon-theme

Install hicolor-icon-theme by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: /usr/share/icons/hicolor

Short Descriptions

/usr/share/icons/hicolor/* contains icon definitions used as defaults

icon-naming-utils-0.8.90

Introduction to icon-naming-utils

The icon-naming-utils package contains a Perl script used for maintaining backwards compatibility with current desktop icon themes, while migrating to the names specified in the [Icon Naming Specification](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://tango.freedesktop.org/releases/icon-naming-utils-0.8.90.tar.bz2>
- Download MD5 sum: dd8108b56130b9eedc4042df634efa66
- Download size: 57 KB
- Estimated disk space required: 440 KB
- Estimated build time: less than 0.1 SBU

icon-naming-utils Dependencies

Required

[XML-Simple-2.25](#)

Installation of icon-naming-utils

Install icon-naming-utils by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: icon-name-mapping

Installed Libraries: None

Installed Directories: /usr/share/dtds and /usr/share/icon-naming-utils

Short Descriptions

`icon-name-mapping` is a Perl script used for maintaining backwards compatibility with current desktop icon themes, while migrating to the names specified in the Icon Naming Specification

lxde-icon-theme-0.5.1

Introduction to LXDE Icon Theme

The LXDE Icon Theme package contains nuoveXT 2.2 Icon Theme for LXDE.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/lxde/lxde-icon-theme-0.5.1.tar.xz>
- Download MD5 sum: 7467133275edbbcc79349379235d4411
- Download size: 4.3 MB
- Estimated disk space required: 18 MB
- Estimated build time: less than 0.1 SBU

LXDE Icon Theme Dependencies

Optional

[GTK+-3.24.43](#) (for `gtk-update-icon-cache` command)

Installation of LXDE Icon Theme

Install LXDE Icon Theme by running the following commands:

```
./configure --prefix=/usr
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have installed one of the optional dependencies, run the following command as the `root` user:

```
gtk-update-icon-cache -qf /usr/share/icons/nuovext2
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: /usr/share/icons/nuovext2

oxygen-icons-6.0.0

Introduction to oxygen-icons

The oxygen icons theme is a photo-realistic icon style, with a high standard of graphics quality.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/oxygen-icons/oxygen-icons-6.0.0.tar.xz>
- Download MD5 sum: 7de8aed8caa14ccf5b4906e68c020d55
- Download size: 231 MB
- Estimated disk space required: 453 MB
- Estimated build time: less than 0.1 SBU

oxygen-icons Dependencies

Required

[extra-cmake-modules-6.5.0](#) and [Qt-6.7.2](#)

Installation of oxygen-icons

First, enable scalable icons:

```
sed -i '/( oxygen/ s)/scalable /' CMakeLists.txt
```

Install oxygen-icons by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr -W no-dev ..
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: none

Installed Libraries: none

Installed Directory: /usr/share/icons/oxygen

Part VII. KDE

KDE is a comprehensive desktop environment with a huge number of applications written for it and a huge amount of users. It is based on the Qt framework.

For more information visit the official KDE project site at <https://kde.org/>.

Chapter 29. Introduction to KDE

KDE Preliminaries

KDE is a comprehensive desktop environment with a huge number of applications written for it and a huge amount of users. It is based on the Qt framework.

The KDE5 system has two main blocks: the libraries, called KDE Frameworks 6 or KF6, which can be used in other environments, and the desktop environment called KDE Plasma 6. Most of the applications written by the KDE team only use KF6, and do not need the plasma environment.

For more information visit the official KDE project site at <https://kde.org/>.

extra-cmake-modules-6.5.0

Introduction to Extra Cmake Modules

The Extra Cmake Modules package contains extra CMake modules used by KDE Frameworks and other packages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/extra-cmake-modules-6.5.0.tar.xz>
- Download MD5 sum: bceea430e534852558b54f4bd3354474
- Download size: 336 KB
- Estimated disk space required: 7.9 MB
- Estimated build time: less than 0.1 SBU

Extra Cmake Modules Dependencies

Required

[CMake-3.30.2](#)

Optional

[sphinx-8.0.2](#) (for building documentation), [PyQt](#) (experimental support for building KDE Python bindings), and [ReuseTool](#) (for running internal tests)

Installation of Extra Cmake Modules

Install Extra Cmake Modules by running the following commands:

```
sed -i '/lib64"/s/64//' kde-modules/KDEInstallDirsCommon.cmake &&

sed -e '/PACKAGE_INIT/i set(SAVE_PACKAGE_PREFIX_DIR "${PACKAGE_PREFIX_DIR}")' \
-e '/^include/a set(PACKAGE_PREFIX_DIR "${SAVE_PACKAGE_PREFIX_DIR}")' \
-i ECMConfig.cmake.in &&

mkdir build &&
cd    build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr .. &&
make
```

This package does not come with a test suite.

Note

Unlike other KF6 packages, this module is installed in /usr because it can be used by some non-KF6 packages.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ... Modules/KDEInstallDirs.cmake`: This command disables applications using `cmake` from attempting to install files in a `lib64` sub-directory.

`sed ... ECMConfig.cmake.in`: This command protects the global `cmake` variable `PACKAGE_PREFIX_DIR` from being changed when checking ECM presence: since we install ECM into `/usr`, the check would set that variable to `/usr`, while most KDE packages expect it to be set to `/opt/kf5` and would fail to build if it is set to something else.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/ECM and /usr/share/doc/ECM (if documentation was built)

Phonon-4.12.0

Introduction to Phonon

Phonon is the multimedia API for KDE. It replaces the old aRts package. Phonon needs the VLC backend.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/phonon/4.12.0/phonon-4.12.0.tar.xz>
- Download MD5 sum: e80e9c73967080016bdb3c0ee514ceab
- Download size: 400 KB
- Estimated disk space required: 23 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

Phonon Dependencies

Required

[CMake-3.30.2](#), [extra-cmake-modules-6.5.0](#), [GLib-2.80.4](#), and [Qt-6.7.2](#)

[phonon-backend-vlc-0.12.0](#) needs to be installed afterwards for multimedia operation in KDE.

Optional

[PulseAudio-17.0](#)

Installation of Phonon

Install Phonon by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D PHONON_BUILD_QT5=OFF \
-W no-dev .. &&
make
```

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply higher level of the compiler optimizations.

`-D PHONON_BUILD_QT5=OFF`: By default both the Qt5 and Qt6 versions of this package are built. We currently just want the Qt6 version, since nothing in BLFS uses the Qt5 version.

Contents

Installed Programs: phononsettings

Installed Libraries: libphonon4qt6.so and libphonon4qt6experimental.so

Installed Directories: /usr/include/phonon4qt6, /usr/lib/cmake/phonon4qt6, and /usr/lib/plugins/designer

Phonon-backend-vlc-0.12.0

Introduction to the Phonon-backend-vlc

This package provides a Phonon backend which utilizes the VLC media framework.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/phonon/phonon-backend-vlc/0.12.0/phonon-backend-vlc-0.12.0.tar.xz>
- Download MD5 sum: 2a27b5d249f97a15040481008fb16e1b
- Download size: 72 KB
- Estimated disk space required: 6.6 MB
- Estimated build time: 0.2 SBU

Phonon-backend-vlc Dependencies

Required

[phonon-4.12.0](#) and [VLC-3.0.21](#) (gui is not needed)

Installation of Phonon-backend-vlc

Install Phonon-backend-vlc by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D PHONON_BUILD_QT5=OFF \
.. &&
make
```

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply higher level of compiler optimizations.

Contents

Installed Program: none

Installed Libraries: phonon_vlc_qt6.so

Installed Directory: /usr/lib/plugins/phonon4qt6_backend

Polkit-Qt-0.200.0

Introduction to Polkit-Qt

Polkit-Qt provides an API to PolicyKit in the Qt environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/polkit-qt-1/polkit-qt-1-0.200.0.tar.xz>
- Download MD5 sum: 6f23bc987ea9c49b4ef21983b9d1dc15
- Download size: 60 KB
- Estimated disk space required: 3.8 MB
- Estimated build time: 0.1 SBU

Polkit-Qt Dependencies

Required

[CMake-3.30.2](#), [Polkit-125](#), and [Qt-6.7.2](#)

Installation of Polkit-Qt

Install Polkit-Qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-D QT_MAJOR_VERSION=6 \
-W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply higher level of the compiler optimizations.

Contents

Installed Programs: none

Installed Libraries: libpolkit-qt6-agent-1.so, libpolkit-qt6-core-1.so, and libpolkit-qt6-gui-1.so

Installed Directories: /usr/include/polkit-qt6-1 and /usr/lib/cmake/PolkitQt6-1

plasma-wayland-protocols-1.13.0

Introduction to the Plasma-wayland-protocols

This package provides a custom set of protocol definitions for KDE.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/plasma-wayland-protocols/plasma-wayland-protocols-1.13.0.tar.xz>
- Download MD5 sum: 5d30de6367ab1ff92b2ece7b159b3c8b
- Download size: 44 KB
- Estimated disk space required: 744 KB
- Estimated build time: less than 0.1 SBU

Plasma-wayland-protocols Dependencies

Required

[extra-cmake-modules-6.5.0](#)

Installation of Plasma-wayland-protocols

Install Plasma-wayland-protocols by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr ..
```

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: none

Installed Libraries: none

Installed Directories: /usr/share/plasma-wayland-protocols and /usr/lib/cmake/PlasmaWaylandProtocols

Chapter 30. KDE Frameworks 6

KDE Frameworks 6 Pre-installation Configuration

KF6 can be installed in `/usr` or `/opt/kf6`. The BLFS editors recommend the latter in the BLFS environment.

Installing in `/usr`

One option is to install KDE Frameworks into the `/usr` hierarchy. This creates a simpler setup but makes it more difficult to try multiple versions of KDE Frameworks.

```
export KF6_PREFIX=/usr
```

It is a good idea to add the following variables to your system or personal profiles:

```
cat >> /etc/profile.d/qt6.sh << "EOF"
# Begin kf6 extension for /etc/profile.d/qt6.sh

pathappend /usr/lib/plugins          QT_PLUGIN_PATH
pathappend $QT6DIR/lib/plugins        QT_PLUGIN_PATH

pathappend /usr/lib/qt6/qml          QML2_IMPORT_PATH
pathappend $QT6DIR/lib/qml           QML2_IMPORT_PATH

# End extension for /etc/profile.d/qt6.sh
EOF

cat > /etc/profile.d/kf6.sh << "EOF"
# Begin /etc/profile.d/kf6.sh

export KF6_PREFIX=/usr

# End /etc/profile.d/kf6.sh
EOF
```

Note

If qt6 was installed in `/usr`, the `$QT6DIR/lib/` portions of the above paths may need to be changed to `$QT6DIR/lib/qt6/`.

Additionally, if [Sudo-1.9.15p5](#) is installed, these variables should be available to the super user. Execute the following commands as the `root` user:

```
cat >> /etc/sudoers.d/qt << "EOF"
Defaults env_keep += QT_PLUGIN_PATH
Defaults env_keep += QML2_IMPORT_PATH
EOF

cat >> /etc/sudoers.d/kde << "EOF"
Defaults env_keep += KF6_PREFIX
EOF
```

Installing in `/opt`

A method of building multiple versions installs KDE Frameworks in the `/opt` hierarchy:

```
export KF6_PREFIX=/opt/kf6
```

If you are not installing KDE Frameworks in `/usr`, you will need to make some additional configuration changes. Best practice is to add those to your system or personal profile:

```
cat > /etc/profile.d/kf6.sh << "EOF"
# Begin /etc/profile.d/kf6.sh

export KF6_PREFIX=/opt/kf6

pathappend $KF6_PREFIX/bin          PATH
pathappend $KF6_PREFIX/lib/pkgconfig PKG_CONFIG_PATH

pathappend $KF6_PREFIX/etc/xdg      XDG_CONFIG_DIRS
pathappend $KF6_PREFIX/share        XDG_DATA_DIRS

pathappend $KF6_PREFIX/lib/plugins   QT_PLUGIN_PATH
pathappend $KF6_PREFIX/lib/plugins/kcms QT_PLUGIN_PATH

pathappend $KF6_PREFIX/lib/qml       QML2_IMPORT_PATH

pathappend $KF6_PREFIX/lib/python3.12/site-packages PYTHONPATH
```

```

pathappend $KF6_PREFIX/share/man          MANPATH
pathappend $KF6_PREFIX/include           CPLUS_INCLUDE_PATH
# End /etc/profile.d/kf6.sh
EOF

cat >> /etc/profile.d/qt6.sh << "EOF"
# Begin Qt6 changes for KF6

pathappend /usr/lib/plugins             QT_PLUGIN_PATH
pathappend $QT6DIR/plugins              QT_PLUGIN_PATH
pathappend $QT6DIR/qml                 QML2_IMPORT_PATH

# End Qt6 changes for KF6
EOF

```

Expand your `/etc/ld.so.conf` file:

```

cat >> /etc/ld.so.conf << "EOF"
# Begin KF6 addition

/opt/kf6/lib

# End KF6 addition
EOF

```

Several KDE Frameworks and KDE Plasma packages install files into D-Bus, Polkit, and systemd directories. When installing KDE 6 in a location other than `/usr`, D-Bus, Polkit, and systemd need to be able to find these files. The easiest way to achieve this is to create the following symlinks (as the `root` user):

```

install -v -dm755      $KF6_PREFIX/{etc,share} &&
ln -sfv /etc/dbus-1    $KF6_PREFIX/etc        &&
ln -sfv /usr/share/dbus-1 $KF6_PREFIX/share    &&
ln -sfv /usr/share/polkit-1 $KF6_PREFIX/share   &&
install -v -dm755      $KF6_PREFIX/lib         &&
ln -sfv /usr/lib/systemd $KF6_PREFIX/lib       &&

```

Some packages may also install icons from the "hicolor" icon set. Since that icon set is used by many packages, it is a good idea to create a symlink to the one in `/usr/share` to avoid having multiple installations of [hicolor-icon-theme-0.18](#). Run the following commands as the `root` user:

```

install -v -dm755          $KF6_PREFIX/share/icons &&
ln -sfv /usr/share/icons/hicolor $KF6_PREFIX/share/icons

```

Tip

Sometimes, the installation paths are hardcoded into installed files. This is the reason why `/opt/kf6` is used as installation prefix instead of `/opt/kf6-6.5.0`. After installing KDE Frameworks, you may rename the directory and create a symlink:

```

mv /opt/kf6{-,6.5.0}
ln -sfv kf6-6.5.0 /opt/kf6

```

Later on, you may want to install other versions of KDE Frameworks. To do that, just remove the symlink and use `/opt/kf6` as the prefix again. Which version of KDE Frameworks you use depends only on where the symlink points. No other reconfiguration will be needed.

Building KDE Frameworks 6.5.0 (KF6)

KDE Frameworks is a collection of libraries based on top of Qt6 and QML derived from the previous KDE libraries. They can be used independent of the KDE Display Environment (Plasma 6).

This package is known to build and work properly using an LFS 12.2 platform.

The instructions below build all of the KDE Frameworks packages in one step by using a bash script.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/>
- Download MD5 sum: See Below
- Download size: 99 MB
- Estimated disk space required: 2.6 GB (183 MB installed)
- Estimated build time: 18 SBU (using parallelism=8)

KF6 Dependencies

Required

[extra-cmake-modules-6.5.0](#), [breeze-icons-6.5.0](#), [docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [libcanberra-0.30](#), [libgcrypt-1.11.0](#), [libical-3.0.18](#), [libxslt-1.1.42](#), [lmbd-0.9.31](#), [qca-2.3.9](#), [qrencode-4.1.1](#), [plasma-wayland-protocols-1.13.0](#), [PyYAML-6.0.2](#), [shared-mime-info-2.4](#), [URI-5.28](#), and [Wget-1.24.5](#) (required to download the packages)

Recommended

[Aspell-0.60.8.1](#) (Dictionary backend for Sonnet), [Avahi-0.8](#) (DNS-SD backend for KDNSSD), [ModemManager-1.18.12](#) (needed to build ModemManager-Qt), [NetworkManager-1.48.8](#) (needed to build NetworkManager-Qt), [polkit-qt-0.200.0](#) (Authentication backend for KAuth), and [Vulkan-Loader-1.3.294](#) (Add support for Vulkan graphics drivers)

Furthermore, the instructions below assume that the environment has been set up as described in [Introduction to KF6](#).

Optional

[BlueZ-5.77](#) (needed to build Bluez-Qt), [Datamatrix](#) (deemed recommended for Prison by upstream)

Runtime dependency for FrameworkIntegration

[Noto fonts](#)

Additional recommended dependencies for kapidox

[Doxygen-1.12.0](#) (Run time), [doxypy.py-0.8.8.7](#), [doxyqml-0.5.3](#), and [requests-2.32.3](#)

Additional image formats support in KImageFormats

[libavif-1.1.1](#), [libjxl-0.10.3](#), [libraw-0.21.2](#), [libheif](#), [OpenEXR](#)

Optional dependencies for Solid

[UDisks-2.10.1](#), [UPower-1.90.4](#) and [media-player-info](#) (runtime)

Optional dependency for KWallet

[GPGME-1.23.2](#), built with C++ bindings (which is the default).

Optional dictionary backends for Sonnet

[Hspell](#) and [Hunspell](#)

Downloading KDE Frameworks

The easiest way to get the KDE Frameworks packages is to use a single `wget` to fetch them all at once:

```
url=https://download.kde.org/stable/frameworks/6.5/
wget -r -nH -nd -A '*.xz' -np $url

The options used here are:
  -r          recurse through child directories
  -nH         disable generation of host-prefixed directories
  -nd        do not create a hierarchy of directories
  -A '*.xz'   just get the *.xz files
  -np        don't get parent directories
```

Setting Package Order

The order of building files is important due to internal dependencies. Create the list of files in the proper order as follows:

```

cat > frameworks-6.5.0.md5 << "EOF"
9f93da02571dc32c0c39db4911546731 attica-6.5.0.tar.xz
#bceea430e534852558b54f4bd3354474 extra-cmake-modules-6.5.0.tar.xz
05e0a1b09e3abb81068e4614f7e9b4a5 kapidox-6.5.0.tar.xz
77c9c2100e36a44714c665644a4773c3 karchive-6.5.0.tar.xz
facb9708090a0eb54a3f6c22f2baa86e kcodecs-6.5.0.tar.xz
698ab95acdc713571f917bb70073e2df kconfig-6.5.0.tar.xz
ab4868788299d29e9016bc57be64b6e6 kcoreaddons-6.5.0.tar.xz
763aa4bc47b19a4b7471eca628be7e8a kdbusaddons-6.5.0.tar.xz
a30dc9bc14f6f2c2316a252e1f6ee7b7 kdnsdd-6.5.0.tar.xz
e23c576829b3a35631b729a3826eb381 kguiaddons-6.5.0.tar.xz
54d5c5dc203772577b6e01e7704f23d2 ki18n-6.5.0.tar.xz
fb7a5aec22e99bc69d4c4d50ee418dd5 kidletime-6.5.0.tar.xz
6751e1b3b08bfd34b7f91f90419796a kimageformats-6.5.0.tar.xz
5fd3404f7197097a23d2db7cd776432e kitemmodels-6.5.0.tar.xz
6b6f3cd08284cef4c7bed3d31c5735a4 kitemviews-6.5.0.tar.xz
0a33080b08df165501d065eb0ccb4d6 kplotting-6.5.0.tar.xz
ff9f1410fe8bb6686cc3bc8e1d352f7 kwidgetsaddons-6.5.0.tar.xz
5631073e18780513a3005159dd73e290 kwindowsystem-6.5.0.tar.xz
579eea31abc156e82cc4b746d212f4d2 networkmanager-qt-6.5.0.tar.xz
712a3c59785bd0de4862cce34de58e4c solid-6.5.0.tar.xz
afelcc49342b0b78993172c8a9e1dc20 sonnet-6.5.0.tar.xz
b9df82b41676d91ea0371bd61e6082d0 threadweaver-6.5.0.tar.xz
952c44490af601ebd58cae56a5c38481 kauth-6.5.0.tar.xz
1db39b1e20b31bbff7d2079e9856b067 kcompletion-6.5.0.tar.xz
104bcce4ba8d7d5936ff4b66762ff0 kcrash-6.5.0.tar.xz
43dfbda305f57ae0cda57091202d034a kdoctools-6.5.0.tar.xz
b39dc81ffbea67c69a1e9e41e6c4acb5 kpty-6.5.0.tar.xz
f721e794c896d48f4143bf8054f471d0 kunitconversion-6.5.0.tar.xz
5f37c1b7b58d13109f9d77f2f47610a0 kcolorscheme-6.5.0.tar.xz
3da8c7755a7207ce332775f7bea879a9 kconfigwidgets-6.5.0.tar.xz
36af7db35be0bdae9d9ae10d67e1b1f1 ksERVICE-6.5.0.tar.xz
51be711487b2d3ef3a35bdd2d3eecd74 kglobalaccel-6.5.0.tar.xz
1bc8471d498ad3de7c144bcac545ab29 kpackage-6.5.0.tar.xz
c12d163df68d8aaee566618c2b886fe0 kdesu-6.5.0.tar.xz
a94b4e76eff97c02fe461e10535cf11 kiconthemes-6.5.0.tar.xz
f75c675a77172fd1cb8eaee62086fd93 knotifications-6.5.0.tar.xz
878a3dc9bb42497d6c6b3f9744372635 kjobwidgets-6.5.0.tar.xz
ea43454373673b7dfdc928609b80fbff1 ktextwidgets-6.5.0.tar.xz
7aef61c37386c5750f83a438fc140453 kxmlgui-6.5.0.tar.xz
11f331d02cd2f542cfa5be32f3c780dd kbookmarks-6.5.0.tar.xz
fe89e3850dfe0200c3ee04e0bcf0ea86 kwallet-6.5.0.tar.xz
dadc7cc6e488d1db9b8dc482e89f07a7 kded-6.5.0.tar.xz
0f2bd915931d6501c212ff3c84433126 kio-6.5.0.tar.xz
7ef47616f65781126b29f3ff7c6637d4 kdeclarative-6.5.0.tar.xz
3478698203e69dc14ca9a8fe58b6263c kcmutils-6.5.0.tar.xz
82ff71c1903a2d205b873e0b04715839 kirigami-6.5.0.tar.xz
39559a047c8da6aba78ba1d619353808 syndication-6.5.0.tar.xz
a6d662cf3920c575bb2fb7541c33f691 knewstuff-6.5.0.tar.xz
cb414562458340f793878c5a2b328af8 frameworkintegration-6.5.0.tar.xz
bad702d88666893c19339eab1b65c9c24 kparts-6.5.0.tar.xz
16874e76c5058e1943d04bc3bf512f03 syntax-highlighting-6.5.0.tar.xz
fbe2608ded4b47a1e7974c92d458596a ktexteditor-6.5.0.tar.xz
7dc07ea6306795a866e4ef62e03b54da modemmanager-qt-6.5.0.tar.xz
ca118c3fa1d01c5e98290475a4ef341d kcontacts-6.5.0.tar.xz
ab5907924e9e35d8f328a66a5fa241ad kpeople-6.5.0.tar.xz
124cf04998f0399a300baa585a01db6b bluez-qt-6.5.0.tar.xz
f5d8507325dce3768b77e2565f3e70d7 kfilmetadata-6.5.0.tar.xz
00951e5fd64e184f5e1d4bc3e71f28a2 baloo-6.5.0.tar.xz
#25c09a753edc62593ecca9f56a34f7ef1 breeze-icons-6.5.0.tar.xz
940fe29e162175c932475ed20f241979 krunner-6.5.0.tar.xz
4598c42a8a4e1a93457cb2aebd392bb2 prison-6.5.0.tar.xz
c2eb5204f9f7b43798566666b06fa3ae qqc2-desktop-style-6.5.0.tar.xz
612f5190aebd9d4c89122113fd97e73e kholidays-6.5.0.tar.xz
9c92e566404c25c2bd142c8c5937441f purpose-6.5.0.tar.xz
5bf4fa5abc19addec2ab47440c0b660e kcalendarcore-6.5.0.tar.xz
2f3150f1162013f6c089dd9c48358a58 kquickcharts-6.5.0.tar.xz
9e3db7659cfe0735461d64bb959a115f knotifyconfig-6.5.0.tar.xz
7801932715454a8f7addba171effcb36c kdav-6.5.0.tar.xz
78b6317709ad47539b4728b22618379f kstatusnotifieritem-6.5.0.tar.xz
7bb7d7f6ff6a4d41e81389dba5982fac ksvg-6.5.0.tar.xz
f24694fc1e66236182cf41f1f8a789e2 ktexttemplate-6.5.0.tar.xz
52e318daedfd8b855e0554f66041535a kuserfeedback-6.5.0.tar.xz
EOF

```

In the above list, notice that some files have been commented out with a hash (#) character.

- The extra-cmake-modules entry has been commented out because it was built earlier in the [Introduction to KDE](#).
- The icon package is covered separately at [breeze-icons-6.5.0](#).

Installation of KDE Frameworks

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

- Run the entire script as the root user (not recommended).
- Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
- Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```
as_root()
{
    if [ $EUID = 0 ]; then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\"\\"
    fi
}

export -f as_root
```

Caution

If installing in `/opt` and there is an existing `/opt/kf6` either as a regular directory or a symbolic link, it should be reinitialized (as `root`):

```
mv -v /opt/kf6 /opt/kf6.old
install -v -dm755           $KF6_PREFIX/{etc,share} &&
ln -sfv /etc/dbus-1         $KF6_PREFIX/etc      &&
ln -sfv /usr/share/dbus-1   $KF6_PREFIX/share    &&
ln -sfv /usr/share/polkit-1 $KF6_PREFIX/share    &&
install -v -dm755           $KF6_PREFIX/lib      &&
ln -sfv /usr/lib/systemd   $KF6_PREFIX/lib      &&
```

First, start a subshell that will exit on error:

```
bash -e
```

Install all of the packages by running the following commands:

```
while read -r line; do

    # Get the file name, ignoring comments and blank lines
    if $(echo $line | grep -E -q '^ *$|^#'); then continue; fi
    file=$(echo $line | cut -d" " -f2)

    pkg=$(echo $file|sed 's|^.*||')          # Remove directory
    packagedir=$(echo $pkg|sed 's|\.\tar.*||') # Package directory

    name=$(echo $pkg|sed 's|-.*$||') # Isolate package name

    tar -xf $file
    pushd $packagedir

    # kapidox is a python module
    case $name in
        kapidox)
            pip3 wheel -w dist --no-build-isolation --no-deps --no-cache-dir $PWD
            as_root pip3 install --no-index --find-links=dist --no-cache-dir --no-user kapidox
            popd
    esac
done
```

```

        rm -rf $packagedir
        continue
        ;;
    esac

    mkdir build
    cd     build

    cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
          -D CMAKE_INSTALL_LIBEXECDIR=libexec \
          -D CMAKE_PREFIX_PATH=$QT6DIR           \
          -D CMAKE_SKIP_INSTALL_RPATH=ON         \
          -D CMAKE_BUILD_TYPE=Release            \
          -D BUILD_TESTING=OFF                  \
          -W no-dev ..

    make
    as_root make install
    popd

    as_root rm -rf $packagedir
    as_root /sbin/ldconfig

done < frameworks-6.5.0.md5

exit

```

Note

Any modules that have been omitted can be installed later by using the same `mkdir build; cd build; cmake; make; make install` procedure as above.

Sometimes the installation paths are hardcoded into installed files. If the installed directory is not /usr, rename the directory and create a symlink:

```

mv -v /opt/kf6 /opt/kf6-6.5.0
ln -sfnv kf6-6.5.0 /opt/kf6

```

Command Explanations

`-D CMAKE_PREFIX_PATH=$QT6DIR`: This switch is used to allow cmake to find the proper Qt libraries.

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to apply higher level of the compiler optimizations.

`-D BUILD_TESTING=OFF`: This switch is used to prevent building test programs and libraries that are of no use to an end user.

Contents

Installed Programs:

balooctl, baloo_file, baloo_file_extractor, baloosearch, balooshow, checkXML6, depdiagram-generate, depdiagram-generate-all, depdiagram-prepare, desktoptojson, gentrigrams, kactivities-cli, kapidox_generate, kate-syntax-highlighter, kbuildsycoca6, kcookiejar6, kdebugdialog6, kded6, kdeinit6, kdeinit6_shutdown, kdeinit6_wrapper, kf6-config, kf6kross, kgendesignerplugin, kglobalaccel6, kiconfinder6, kjs6, kjscmd6, kjsconsole, knewstuff-dialog, kpackagelauncherqml, kpackagetool6, kquitapp6, kreadconfig6, kshell6, ktelnetservice6, ktrash6, kwalletd6, kwallet-query, kwrapper6, kwriteconfig6, meinproc6, parsetrigrams, plasmapkg2, preparetips6, protocoltojson, and solid-hardware6

Installed Libraries:

libkdeinit6_klauncher.so, libKF6Activities.so, libKF6ActivitiesStats.so, libKF6Archive.so, libKF6Attica.so, libKF6AuthCore.so, libKF6Auth.so, libKF6Baloo.so, libKF6BluezQt.so, libKF6Bookmarks.so, libKF6CalendarCore.so, libKF6CalendarEvents.so, libKF6Codecs.so, libKF6Completion.so, libKF6ConfigCore.so, libKF6ConfigGui.so, libKF6ConfigWidgets.so, libKF6Contacts.so, libKF6CoreAddons.so, libKF6Crash.so, libKF6DAV.so, libKF6DBusAddons.so, libKF6Declarative.so, libKF6DNSSD.so, libKF6DocTools.so, libKF6Emoticons.so, libKF6FileMetaData.so, libKF6GlobalAccel.so, libKF6GuiAddons.so, libKF6Holidays.so, libKF6I18n.so, libKF6IconThemes.so, libKF6IdleTime.so, libKF6ItemModels.so, libKF6ItemViews.so, libKF6JobWidgets.so, libKF6JSApi.so, libKF6JsEmbed.so, libKF6JS.so, libKF6KCMUtils.so, libKF6KDELibs4Support.so, libKF6KHTML.so, libKF6KIOCore.so, libKF6KIOFileWidgets.so, libKF6KIOGui.so, libKF6KIONTLM.so, libKF6KIOWidgets.so, libKF6Kirigami2.so, libKF6KrossCore.so, libKF6KrossUi.so, libKF6MediaPlayer.so, libKF6NetworkManagerQt.so, libKF6NewStuffCore.so, libKF6NewStuff.so, libKF6Notifications.so, libKF6NotifyConfig.so, libKF6Package.so, libKF6Parts.so, libKF6PeopleBackend.so, libKF6People.so, libKF6PeopleWidgets.so, libKF6PlasmaQuick.so, libKF6Plasma.so, libKF6Plotting.so, libKF6Prison.so, libKF6Pty.so, libKF6Purpose.so, libKF6PurposeWidgets.so, libKF6QuickAddons.so, libKF6Runner.so, libKF6Service.so, libKF6Solid.so, libKF6SonnetCore.so, libKF6SonnetUi.so, libKF6Style.so, libKF6Su.so,

libKF6SyntaxHighlighting.so, libKF6TextEditor.so, libKF6TextWidgets.so, libKF6ThreadWeaver.so, libKF6UnitConversion.so, libKF6Wallet.so, libKF6WaylandClient.so, libKF6WaylandServer.so, libKF6WidgetsAddons.so, libKF6WindowSystem.so, libKF6XmlGui.so, libKF6XmlRpcClient.so, and libkwalletbackend6.so

Installed Directories: /opt/kf6 (symlink to /opt/kf6-6.5.0) if installing in /opt

Short Descriptions

checkXML6	is a tool to check for syntax errors in KDE DocBook XML files
depdiagram-generate	is a tool to generate a dependency diagram
depdiagram-generate-all	is a tool to generate a dependency diagram for all frameworks at once
depdiagram-prepare	is a tool to prepare dot files
desktoptojson	is a tool to convert a .desktop file to a .json file
kbuildsycoca6	rebuilds the KService desktop file system configuration cache
kcookiejar6	is a command line interface to the HTTP cookie store used by KDE, a D-BUS service to store/retrieve/clean cookies
kded6	consolidates several small services in one process
kdeinit6	is a process launcher somewhat similar to the famous init used for booting UNIX
kf6-config	is a command line program used to retrieve information about KDE installation or user paths
kf6kross	executes kross scripts written in KDE Javascript, Python, Ruby, Java and Falcon
kgendesignerplugin	generates widget plugins for Qt(TM) Designer
kglobalaccel6	is a daemon use to register the key bindings and for getting notified when the action triggered
kjs6	is KDE ECMAScript/JavaScript engine
kjscmd6	is a tool for launching KJSEmbed scripts from the command line
kjsconsole	is a console for <code>kjs6</code>
kpackagelauncherqml	is a commandline tool for launching kpackage QML application
kpackage tool6	is a command line kpackage tool
kreadconfig6	is a command line tool to retrieve values from KDE configuration files
kshell6	start applications via kdeinit
ktelnetservice6	is a telnet service
ktrash6	is a helper program to handle the KDE trash can
kwalletd6	is the wallet manager daemon
kwriteconfig6	is a command line tool to write values in KDE configuration files
meinproc6	converts DocBook files to HTML
plasmapkg2	is a tool to install, list, remove Plasma packages
preparertips6	is a script to extract the text from a tips file
solid-hardware6	is a command line tool to investigate available devices

Chapter 31. KDE Frameworks 6 Based Applications

ark-24.08.0

Introduction to Ark

The Ark package is a KF6 archive tool. It is a graphical front end to tar and similar tools.

This package is known to build and work properly using LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/ark-24.08.0.tar.xz>
- Download MD5 sum: daec28b797eb35c840fd0d43dad5ab00
- Download size: 2.9 MB
- Estimated disk space required: 38 MB

- Estimated build time: 0.4 SBU (using parallelism=4)

Ark Dependencies

Required

[KDE Frameworks-6.5.0](#) and [libarchive-3.7.4](#)

Recommended

[cpio-2.15](#), [p7zip-17.04](#), [UnRar-7.0.9](#), [UnZip-6.0](#), and [Zip-3.0](#)

Installation of Ark

Install Ark by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: ark

Installed Libraries: libkerfuffle.so

Installed Directories: \$KF6_PREFIX/lib/plugins/kerfuffle and \$KF6_PREFIX/share/doc/HTML/*/ark

Short Descriptions

`ark` is a KF6-based Archive Manager

dolphin-24.08.0

Introduction to dolphin

The dolphin package is a KF6 file manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/dolphin-24.08.0.tar.xz>
- Download MD5 sum: 33f91651409e8e59906caf001fd75eac
- Download size: 5.4 MB
- Estimated disk space required: 78 MB
- Estimated build time: 0.7 SBU (using parallelism=4)

dolphin Dependencies

Required

[KDE Frameworks-6.5.0](#)

Installation of dolphin

Install dolphin by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: dolphin

Installed Libraries: libdolphinvcs.so and libdolphinprivate.so

Installed Directories: \$KF6_PREFIX/include/Dolphin, \$KF6_PREFIX/lib/cmake/DolphinVcs, \$KF6_PREFIX/lib/plugins/dolphin, and \$KF6_PREFIX/share/doc/HTML/*/dolphin

Short Descriptions

`dolphin` is a KF6-based file manager

dolphin-plugins-24.08.0

Introduction to dolphin-plugins

The dolphin-plugins package provides extra plugins for the dolphin file manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/dolphin-plugins-24.08.0.tar.xz>
- Download MD5 sum: 6d1ca867d047e2683cfa4f40cb66df56
- Download size: 312 KB
- Estimated disk space required: 23 MB
- Estimated build time: 0.2 SBU (using parallelism=4)

dolphin-plugins Dependencies

Required

[dolphin-24.08.0](#)

Installation of dolphin-plugins

Install dolphin-plugins by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: In \$KF6_PREFIX/lib/plugins/dolphin/vcs/: fileviewsvnplugin.so, fileviewgitplugin.so, fileviewbazaarplugin.so, fileviewdropboxplugin.so, fileviewhgplugin.so, makefileactions.so, and mountisoaction.so

Installed Directories: None

kdenlive-24.08.0

Introduction to Kdenlive

The Kdenlive package is a KF6 based video editor.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/kdenlive-24.08.0.tar.xz>
- Download MD5 sum: 4706c36669178cae1b8248bf53fb921a
- Download size: 12 MB
- Estimated disk space required: 225 MB
- Estimated build time: 1.7 SBU (using parallelism=4)

Kdenlive Dependencies

Required

[KDE Frameworks-6.5.0](#), [mlt-7.26.0](#), and [y4l-utils-1.28.1](#) (runtime)

Recommended

[breeze-icons-6.5.0](#)

Installation of Kdenlive

Install Kdenlive by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release          \
      -D BUILD_TESTING=OFF                \
      -W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: kdenlive and kdenlive_render

Installed Libraries: mltpreview.so

Installed Directories: \$KF6_PREFIX/share/doc/HTML/*/kdenlive and \$KF6_PREFIX/share/kdenlive

Short Descriptions

`kdenlive`

is an open source non-linear video editor which supports a large number of formats

`kdenlive_render` is a render program for `kdenlive`

khelpcenter-24.08.0

Introduction to Khelpcenter

Khelpcenter is an application to show KDE Applications' documentation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/khelpcenter-24.08.0.tar.xz>
- Download MD5 sum: db5ffba114392ca66db1c2733f2668ed
- Download size: 4.0 MB
- Estimated disk space required: 22 MB
- Estimated build time: 0.2 SBU (using parallelism=4)

Khelpcenter Dependencies

Required

[KDE Frameworks-6.5.0](#), [libxml2-2.13.3](#), and [xapian-1.4.26](#)

Recommended

[qtwebengine-6.7.2](#)

Installation of Khelpcenter

Install khelpcenter by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: khelpcenter

Installed Libraries: None

Installed Directories: \$KF6_PREFIX/share/khelpcenter,
\$KF6_PREFIX/share/doc/HTML/*/{khelpcenter,fundamentals,onlinehelp}

Short Descriptions

`khelpcenter` is the help viewer for KDE applications

Konsole-24.08.0

Introduction to Konsole

The Konsole package is a KF6 based terminal emulator.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/konsole-24.08.0.tar.xz>
- Download MD5 sum: 9cce74d7e4bddce4d574c021dd3d8139
- Download size: 1.7 MB
- Estimated disk space required: 61 MB
- Estimated build time: 0.6 SBU (using parallelism=4)

Additional Downloads

- Optional patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/konsole-24.08.0-scrollbar-1.patch>

Konsole Dependencies

Required

[KDE Frameworks-6.5.0](#)

Installation of Konsole

Note

In non-plasma environments, the konsole scrollbar and its handle do not show up well. If desired, apply the optional patch to the package:

```
patch -Np1 -i ../konsole-24.08.0-scrollbar-1.patch
```

The patch makes the scrollbar light gray with a small white border. The handle is dark gray. If desired, the colors may be changed to taste by editing the patch.

Install Konsole by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: konsole and konsoleprofile

Installed Libraries: libkdeinit6_konsole.so and libkonsoleprivate.so

Installed Directories: \$KF6_PREFIX/share/doc/HTML/*/konsole, \$KF6_PREFIX/share/konsole, and
\$KF6_PREFIX/share/kxmlgui6/konsole

Short Descriptions

<code>konsole</code>	is an X terminal emulation which provides a command-line interface
<code>konsoleprofile</code>	is a command-line tool to change the current tab's profile options

Introduction to konversation

The konversation package is a KF6 based IRC client.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/konversation-24.08.0.tar.xz>
- Download MD5 sum: bcbecd7ec3035e6fbe559b9b01c0d38
- Download size: 4.2 MB
- Estimated disk space required: 69 MB
- Estimated build time: 0.6 SBU (using parallelism=4)

konversation Dependencies

Required

[KDE Frameworks-6.5.0](#)

Installation of konversation

Install konversation by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: konversation

Installed Libraries: None

Installed Directories: \$KF6_PREFIX/share/doc/HTML/*/konversation, and \$KF6_PREFIX/share/konversation

Short Descriptions

`konversation` is an IRC client

libkexiv2-24.08.0

Introduction to libkexiv2

Libkexiv2 is a KDE wrapper around the Exiv2 library for manipulating image metadata.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/libkexiv2-24.08.0.tar.xz>
- Download MD5 sum: 811fb2ce8736e93790d8f6f1b20f6d09
- Download size: 60 KB
- Estimated disk space required: 3.6 MB
- Estimated build time: 0.2 SBU

libkexiv2 Dependencies

Required

[KDE Frameworks-6.5.0](#) and [Exiv2-0.28.3](#)

Installation of libkexiv2

Install libkexiv2 by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_WITH_QT6=ON \
-D BUILD_TESTING=OFF \
-W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libKExiv2Qt6.so

Installed Directories: \$KF5_PREFIX/include/KExiv2Qt6 and \$KF5_PREFIX/lib/cmake/KExiv2Qt6

plasma-activities-6.1.4

Introduction to plasma-activities-6.1.4

The plasma-activities-6.1.4 application provides core components for KDE Activities. It is normally built with [Plasma-6.1.4](#) but is included here because it is needed for [okular-24.08.0](#) if [Plasma-6.1.4](#) is not yet built.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/plasma/6.1.4/plasma-activities-6.1.4.tar.xz>
- Download MD5 sum: 250385889199481e81b979bd0dd51669
- Download size: 72 KB
- Estimated disk space required: 8.0 MB
- Estimated build time: 0.2 SBU (using parallelism=4)

plasma-activities Dependencies

Required

[Boost-1.86.0](#) and [KDE Frameworks-6.5.0](#)

Installation of plasma-activities

Install plasma-activities by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \

```

```
-W no-dev .. &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: plasma-activities-cli6

Installed Libraries: libPlasmaActivities.so

Installed Directories: \$KF6_PREFIX/include/PlasmaActivities, \$KF6_PREFIX/lib/cmake/PlasmaActivities, and \$KF6_PREFIX/share/share/qlogging-categories6

Short Descriptions

`plasma-activities-cli6` is the activity manager for KDE applications

okular-24.08.0

Introduction to Okular

Okular is a document viewer for KDE. It can view documents of many types including PDF, PostScript, TIFF, DjVu, DVI, XPS, and ePub.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/okular-24.08.0.tar.xz>
- Download MD5 sum: c216077cd98e25256a1a13e16eea5f11
- Download size: 7.7 MB
- Estimated disk space required: 90 MB
- Estimated build time: 0.8 SBU (using parallelism=4)

Okular Dependencies

Required

[KDE Frameworks-6.5.0](#) and [plasma-activities-6.1.4](#)

Recommended

[libkexiv2-24.08.0](#), [libtiff-4.6.0](#), and [Poppler-24.08.0](#) (built with Qt6, required for PDF support)

Optional

[qca-2.3.9](#), [discount](#), [DjVuLibre](#), [libspectre](#), [libepub](#), and [LibZip](#)

Installation of Okular

Install Okular by running the following commands:

Note

If any of the optional dependencies are installed, remove the associated reference in the SKIP_OPTIONAL environment variable.

```
mkdir build &&  
cd build &&
```

```
SKIP_OPTIONAL='Discount;DjVuLibre;EPub;LibSpectre;LibZip'

cmake -D CMAKE_INSTALL_PREFIX=$KF5_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-D FORCE_NOT_REQUIRED_DEPENDENCIES="$SKIP_OPTIONAL" \
-W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: okular

Installed Library: Okular6Core.so

Installed Directories: \$KF6_PREFIX/include/okular, \$KF6_PREFIX/lib/cmake/Okular6, \$KF6_PREFIX/lib/plugins/okular, \$KF6_PREFIX/share/okular, and \$KF6_PREFIX/share/doc/HTML/*/okular

Short Descriptions

`okular` is a document viewer

libkdrawing-24.08.0

Introduction to libkdrawing

Libkdrawing is a KDE wrapper around the [libraw-0.21.2](#) library for manipulating image metadata.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/libkdrawing-24.08.0.tar.xz>
- Download MD5 sum: b9df8a737f97bad34ae15602c8933ce9
- Download size: 40 KB
- Estimated disk space required: 1.7 MB
- Estimated build time: less than 0.1 SBU

libkdrawing Dependencies

Required

[KDE Frameworks-6.5.0](#) and [libraw-0.21.2](#)

Installation of libkdrawing

Install libkdrawing by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-D QT_MAJOR_VERSION=6 \
-W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libKDcrawQt6.so

Installed Directories: \$KF5_PREFIX/include/KDcrawQt6 and \$KF5_PREFIX/lib/cmake/KDcrawQt6

gwenview-24.08.0

Introduction to Gwenview

Gwenview is a fast and easy-to-use image viewer for KDE.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/gwenview-24.08.0.tar.xz>
- Download MD5 sum: 0307805ec62640cb8bfd645990e834e
- Download size: 6.2 MB
- Estimated disk space required: 56 MB
- Estimated build time: 0.6 SBU (using parallelism=4)

Gwenview Dependencies

Required

[Exiv2-0.28.3](#), [kimageannotator-0.7.1](#), [KDE Frameworks-6.5.0](#), and [Little CMS-2.16](#)

Recommended

[libkdccraw-24.08.0](#)

Optional

[plasma-activities-6.1.4](#) and [CFitsio](#)

Installation of Gwenview

Install Gwenview by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-W no-dev ... &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: gwenview and gwenview_importer

Installed Libraries: libgwenviewlib.so and gvpart.so

Installed Directories: \$KF6_PREFIX/share/{gwenview,doc/HTML/*}/gwenview}

Short Descriptions

`gwenview` is the KDE image viewer
`gwenview_importer` is a Photo importer

libkcddb-24.08.0

Introduction to libkcddb

The libkcddb package contains a library used to retrieve audio CD meta data from the internet.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/libkcddb-24.08.0.tar.xz>
- Download MD5 sum: 22e2c5c7aff55e918254b21d9530ad4c
- Download size: 440 KB
- Estimated disk space required: 11 MB
- Estimated build time: 0.3 SBU

libkcddb Dependencies

Required

[KDE Frameworks-6.5.0](#)

Recommended

[libmusicbrainz-5.1.0](#)

Installation of libkcddb

Install libkcddb by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -D QT_MAJOR_VERSION=6 \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libKCddb6.so

Installed Directories: \$KF6_PREFIX/include/KCddb6 \$KF6_PREFIX/lib/cmake/KCddb6
\$KF6_PREFIX/share/doc/HTML/*/kcontrol

Short Descriptions

`libKCddb6.so` contains functions used to retrieve audio CD meta data from the internet

k3b-24.08.0

Introduction to K3b

The K3b package contains a KF6-based graphical interface to the Cdrtools and dvd+rw-tools CD/DVD manipulation tools. It also combines the capabilities of many other multimedia packages into one central interface to provide a simple-to-operate application that can be used to handle many of your CD/DVD recording and formatting requirements. It is used for creating audio, data, video and mixed-mode CDs as well as copying, ripping and burning CDs and DVDs.

Though k3b can be used to copy almost any DVD to similar medium, it does not provide a way to copy, or reproduce a double-layer DVD onto single-layer medium. Of course, there is not a program anywhere on any platform that can make an exact duplicate of a double-layer DVD onto a single-layer disk, there are programs on some platforms that can compress the data on a double-layer DVD to fit on a single-layer DVD producing a duplicate, but compressed, image. If you need to copy the contents of a double-layer DVD to single-layer medium, you may want to look at the [RMLCopyDVD](#) package.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/k3b-24.08.0.tar.xz>
- Download MD5 sum: a8c61382247d68d6f339a1d52ac28922
- Download size: 10 MB
- Estimated disk space required: 139 MB
- Estimated build time: 1.3 SBU (using parallelism=4)

K3b Dependencies

Required

[KDE Frameworks-6.5.0](#), [libkcddb-24.08.0](#), [libsamplerate-0.2.2](#), and [shared-mime-info-2.4](#)

The CD/DVD drive is detected at run time using [UDisks-2.10.1](#), which must therefore be installed before running `k3b`.

Recommended

[libburn-1.5.6](#), [libdvread-6.1.3](#), and [taglib-2.0.1](#)

There are programs from three packages that k3b will look for at runtime: [Cdrtools-3.02a09](#) (required to burn CD-ROM media), [dvd+rw-tools-7.1](#) (required to burn or format DVD media), and [Cdrdao-1.2.4](#) (required to burn CD-ROM media in DAO (Disk At Once) mode). If you don't need the capability provided by any of the three packages, you don't have to install it. However, a warning message will be generated every time you run the `k3b` program if any are not installed.

Optional Runtime

[FFmpeg-7.0.2](#)

Optional

[FLAC-1.4.3](#), [LAME-3.100](#), [libmad-0.15.1b](#), [libsndfile-1.2.2](#), [libvorbis-1.3.7](#), and [Musepack \(libmpcdec\)](#).

Installation of K3b

Install K3b by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `k3b`

Installed Libraries: `libk3bdevice.so`, `libk3blib.so`, and numerous plugin modules for the installed dependencies

Installed Directories: \$KF6_PREFIX/share/k3b and \$KF6_PREFIX/share/doc/HTML/*/k3b

Short Descriptions

k3b is the graphical CD/DVD program

Further KDE packages

This section did not provide instructions for all of the available packages in the KDE Software Compilation (KDE Gear). The included packages were selected based on what most people would want to use on a standard desktop computer.

For a full list of available packages, have a look on the KDE5 servers at <https://download.kde.org/stable/release-service/24.08.0/src>.

Editor Notes: https://wiki.linuxfromscratch.org/blfs/wiki/Further_KDE_Packages

Some additional packages worth mentioning are:

- Juk: A lightweight music player.
- Dragon: A video player.
- Kcalc: A scientific calculator.
- Kwalletmanager: A credentials management application.
- Marble: A global map program.
- A variety (about 40) of KDE games including: kpat, kfourinline, and kmines.

Note

Some packages will require additional dependencies. To determine what dependencies are needed, run from the top of the source tree:

```
cmake -W no-dev -L .
```

Most of these packages can be built with the standard KDE5 instructions:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev ... &&
make
```

And as the `root` user:

```
make install
```

Chapter 32. KDE Plasma

KDE Plasma is a desktop environment built using KDE Frameworks and QML which runs on top of a fully hardware accelerated graphics stack using Qt, QtQuick, and an OpenGL(-ES) scenegraph.

No additional configuration is needed before building the KDE Plasma because it uses the same configuration that KDE Frameworks uses.

kirigami-addons-1.4.0

Introduction to kirigami-addons

This package provides add-on qml files for the Kirigami framework.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/kirigami-addons/kirigami-addons-1.4.0.tar.xz>
- Download MD5 sum: 27d23279ee0ad5252a862c2671bc05ad
- Download size: 1.3 MB
- Estimated disk space required: 40 MB
- Estimated build time: 0.6 SBU (Using parallelism=4)

kirigami-addons Dependencies

Required

[KDE Frameworks-6.5.0](#)

Recommended

[Vulkan-Loader-1.3.294](#)

Installation of kirigami-addons

Install kirigami-addons by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_BUILD_TYPE=Release          \
      -D BUILD_TESTING=OFF                \
      ..                                  \
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: \$KF6_PREFIX/lib/qml/org/kde/kirigamiaddons and \$KF6_PREFIX/lib/cmake/KF6KirigamiAddons

pulseaudio-qt-1.5.0

Introduction to pulseaudio-qt

This package is a Qt-style wrapper for [PulseAudio-17.0](#). It allows querying and manipulation of various PulseAudio objects such as Sinks, Sources and Streams. It does not wrap the full feature set of libpulse.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/pulseaudio-qt/pulseaudio-qt-1.5.0.tar.xz>
- Download MD5 sum: 9e8c7cc600937f0a21709f87a252c795
- Download size: 36 KB
- Estimated disk space required: 5.7 MB
- Estimated build time: less than 0.1 SBU

Konsole Dependencies

Required

Installation of pulseaudio-qt

Install pulseaudio-qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
-D CMAKE_PREFIX_PATH=$QTDIR \
-D CMAKE_SKIP_INSTALL_RPATH=ON \
-D CMAKE_BUILD_TYPE=Release \
-D BUILD_TESTING=OFF \
-D QT_MAJOR_VERSION=6 \
.. &&
make
```

Now as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: libKF6PulseAudioQt.so

Installed Directories: \$KF6_PREFIX/lib/cmake/KF6PulseAudioQt and \$KF6_PREFIX/include/KF6/KF6PulseAudioQt

Building Plasma

KDE Plasma is a collection of packages based on top of KDE Frameworks and QML. They implement the KDE Display Environment (Plasma).

This package is known to build and work properly using an LFS 12.2 platform.

The instructions below build all of the Plasma packages in one step by using a bash script.

Package Information

- Download (HTTP): <https://download.kde.org/stable/plasma/6.1.4>
- Download MD5 sum: See Below
- Download size: 220 MB
- Estimated disk space required: 1.0 GB (444 MB installed)
- Estimated build time: 14 SBU (using parallelism=8)

Plasma Dependencies

Required

[Boost-1.86.0](#), [GTK+-3.24.43](#), [KDE Frameworks-6.5.0](#), [kirigami-addons-1.4.0](#), [libdisplay-info-0.2.0](#), [libpwquality-1.4.5](#), [libqalculate-5.2.0](#), [libnl-3.10.0](#), [libxcvt-0.1.2](#), [libxkbcommon-1.7.0](#), [Mesa-24.1.5](#) built with [Wayland-1.23.0](#), [pipewire-1.2.3](#), [pulseaudio-qt-1.5.0](#), [qca-2.3.9](#), [qcoro-0.10.0](#), [sassc-3.6.2](#), [taglib-2.0.1](#), [xdotool-3.20211022.1](#), and [Xorg_Evdev Driver-2.10.6](#)

Recommended

[gsettings-desktop-schemas-46.1](#), [libcanberra-0.30](#), [libinput-1.26.1](#), [libpcap-1.10.4](#), [Linux-PAM-1.6.1](#), [lm-sensors-3-6-0](#), [oxygen-icons-6.0.0](#), [pciutils-3.13.0](#), [power-profiles-daemon-0.21](#), and the following Python modules: [psutil-6.0.0](#), [pygdbmi-0.11.0.0](#), [sentry-sdk-2.13.0](#), [urllib3-2.2.2](#) (if they are not installed, they will be downloaded and installed by the drkonqi build procedure)

Recommended (runtime)

[AccountsService-23.13.9](#), [breeze-icons-6.5.0](#), [smartmontools-7.4](#), [xdg-desktop-portal-1.18.2](#), and [Xwayland-24.1.2](#)

Optional

[AppStream-1.0.3](#) (build with -qt=true), [GLU-9.0.3](#), [ibus-1.5.30](#), [qtwebengine-6.7.2](#), [Xorg Synaptics Driver-1.9.2](#), [KDevPlatform](#), [libgps](#), [libhybris](#), [packagekit-qt](#), [Qapt](#), [SCIM](#), and [socat](#) (for pam_kwallet)

Downloading KDE Plasma

The easiest way to get the KDE Plasma packages is to use a single `wget` to fetch them all at once:

```
url=https://download.kde.org/stable/plasma/6.1.4/
wget -r -nH -nd -A '*.xz' -np $url

The options used here are:
-r          recurse through child directories
-nH         disable generation of host-prefixed directories
-nd        do not create a hierarchy of directories
-A '*.xz'   just get the *.xz files
-np        don't get parent directories
```

Setting Package Order

The order of building files is important due to internal dependencies. Create the list of files in the proper order as follows:

```
cat > plasma-6.1.4.md5 << "EOF"
be3516c192fcdb0d7454de4b379f5db  kdecoration-6.1.4.tar.xz
dfd2bd982b5ab4daa5a3545e54cf9cb3  libkscreen-6.1.4.tar.xz
c05c1623be161c149a61ac3db0ce5382  libksysguard-6.1.4.tar.xz
7035691a6f63b28f389d76002afc3da8  breeze-6.1.4.tar.xz
9d82aff77358f55069020a408987c111  breeze-gtk-6.1.4.tar.xz
2e3a5220acf76e2c0edf9701978e802b  layer-shell-qt-6.1.4.tar.xz
250385889199481e81b979bd0dd51669  plasma-activities-6.1.4.tar.xz
d6686b711c1e52c0de7c75b66c9dc6e3  libplasma-6.1.4.tar.xz
98101a610670cfae5779bc9940d8561f  kscreenlocker-6.1.4.tar.xz
ca7bf1418903745c65dcfb7583299ceb  kinfocenter-6.1.4.tar.xz
4c4b58890f3bf7bf1b1d99e89f9271a8  kglobacceld-6.1.4.tar.xz
d297f096ac9492262b8274a024618bd3  kwayland-6.1.4.tar.xz
2d4516080f370a6cf40351266277b010  kwin-6.1.4.tar.xz
131d3250d9124c8f21751f5e5255a7d6  plasma5support-6.1.4.tar.xz
fc8fe31088cad9f1a904554956df708b  plasma-activities-stats-6.1.4.tar.xz
85a3ff80ec16299dc8aa12190ce2bc24  kpiewire-6.1.4.tar.xz
575c902a134bb24214f76c500460df39  plasma-workspace-6.1.4.tar.xz
d000e0ee0f0e530c9dc64ab9252bc17a  plasma-disks-6.1.4.tar.xz
180136a6b20736c359b999cbd3509afc  bluedevil-6.1.4.tar.xz
3f065b337807c2a37d27a90bbcfd8d1  kde-gtk-config-6.1.4.tar.xz
bcfa7f92df7c2f1a6c7718a12f81734c  kmenedit-6.1.4.tar.xz
08590284bc4613804563db94a68c7cd2  kscreenc-6.1.4.tar.xz
801318aea037e3d4f0aa7c9ca4fef8b  kwallet-pam-6.1.4.tar.xz
aa6bc381778b4e10c9798c3abeeb5692  kwrited-6.1.4.tar.xz
9c4f88ca9936252f62705f27687af764  milou-6.1.4.tar.xz
cbeb7b6b0094e9ec19af07ea458c8eb3  plasma-nm-6.1.4.tar.xz
aa32529c005706d2253333c93faa45de  plasma-pa-6.1.4.tar.xz
4c7484c0d5e4352372d32a340656abec  plasma-workspace-wallpapers-6.1.4.tar.xz
f904e90a9db900551d228f20dac06a45  polkit-kde-agent-1-6.1.4.tar.xz
312fe34839a8d0861cc676766a4317e4  powerdevil-6.1.4.tar.xz
adc8dad07c1157e79aed7476025b6925  plasma-desktop-6.1.4.tar.xz
cfbc61347436b9ff7ff43b7130d7955a  kgamma-6.1.4.tar.xz
41d6254b23061c4b4067b0d2db150cbf  ksshaskpass-6.1.4.tar.xz
#7bf7fd0d3a4a381148885cd058872212  plasma-sdk-6.1.4.tar.xz
celbbaf7df576f6ddcc1ladda545657a  sddm-kcm-6.1.4.tar.xz
#5482b86089286d9b0153f09752d57574  discover-6.1.4.tar.xz
#d33ef85fb879f3073f1f26f1695e72f3  breeze-grub-6.1.4.tar.xz
#42bb6940246062b24ee80ac6e3b40e2d  breeze-plymouth-6.1.4.tar.xz
e458fdbb447b0c76e8e49bb772b423a9  kactivitymanagerd-6.1.4.tar.xz
e0b3c21962456a5f9f79a739c477c22c  plasma-integration-6.1.4.tar.xz
#bebea20d8972e4052690877455aea645  plymouth-kcm-6.1.4.tar.xz
6bf11a27e32cc81e689867ddf59080c6  xdg-desktop-portal-dde-6.1.4.tar.xz
96d1d51407b85d22f01e6438e8e492cb  drkonqi-6.1.4.tar.xz
34627cfef9b00bc58bc853790e978dfd  plasma-vault-6.1.4.tar.xz
#d7984e9a5d3c4ac7a6e36b656cc16f3a  plasma-browser-integration-6.1.4.tar.xz
626ef7f2e373085f948d0809413b0d49f  kde-cli-tools-6.1.4.tar.xz
5ba8376be2db662750ef48f582f22352  systemsettings-6.1.4.tar.xz
db8923a9b87e95dd5ec82a20da4ae95b  plasma-thunderbolt-6.1.4.tar.xz
#0106b72c6a46c966d31b7a4afdb456b3  plasma-mobile-6.1.4.tar.xz
#8a7de2aa288ae50190fc86a7d72cbcb  plasma-nano-6.1.4.tar.xz
3bc5c58b1d06e4f4f1667799219ae887  plasma-firewall-6.1.4.tar.xz
bf76a58afb8def0a0d07ea65429154fc  plasma-systemmonitor-6.1.4.tar.xz
e6e3a3bc681b7044b5219ed5417f0310  qqc2-breeze-style-6.1.4.tar.xz
```

```

d3096345a478a821fa16620dadf95fb4  ksystemstats-6.1.4.tar.xz
03a717ac104f6f44ab9dffafdc37710b  oxygen-sounds-6.1.4.tar.xz
a2f8684825ee231fefad28589f321ce7  kdeplasma-addons-6.1.4.tar.xz
#4691fcce4a83314e82c8d1b64bf23c8e  flatpak-kcm-6.1.4.tar.xz
8175024493047e4e98cd12f4d52bc24b  plasma-welcome-6.1.4.tar.xz
8431d1bf294e2ba3cacd9c455678d118  ocean-sound-theme-6.1.4.tar.xz
ed8102ad60b2c8ae9ce12dcff3cd5  print-manager-6.1.4.tar.xz
#523c82bcf113168b2113612d7b20e544  wacomtablet-6.1.4.tar.xz
#ebb06249b835ea73743c9cf6e91df4df  kwayland-integration-6.1.4.tar.xz
#87332ae4b1c87a6921651ae9211a7e86  krdp-6.1.4.tar.xz
7ca2419cbdb9e4e9b2822b930ac5ec18  oxygen-6.1.4.tar.xz
EOF

```

About Commented Out Packages

In the above list, several files are commented out with a hash (#) character.

- The plasma-sdk package is optional and used for software development.
- The discover package requires [AppStream-1.0.3](#) to be built with the -D qt=true switch.
- The breeze-grub, breeze-plymouth, and plymouth-kcm packages above are all for customized support of [Plymouth](#) which is designed to be run within an initial ram disk during boot (see [the section called "About initramfs"](#)).
- The plasma-browser-integration and krdp-6.1.4.tar.xz packages are only used by plasma-meta which is not installed by BLFS.
- The plasma-nano package is used for embedded systems.
- : The plasma-mobile package provides phone functionality for Plasma.
- The flatpak-kcm package is for managing support of flatpak applications.
- The wacomtablet and kwayland-integration applications require plasma5 support.

Installation of Plasma

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

- Run the entire script as the root user (not recommended).
- Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
- Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```

as_root()
{
    if [ $EUID = 0 ];      then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\\"
    fi
}

export -f as_root

```

First, start a subshell that will exit on error:

```
bash -e
```

Install all of the packages by running the following commands:

```

while read -r line; do

    # Get the file name, ignoring comments and blank lines
    if $(echo $line | grep -E -q '^ *$|^#'); then continue; fi

```

```

file=$(echo $line | cut -d" " -f2)

pkg=$(echo $file|sed 's|^.*|||')           # Remove directory
packagedir=$(echo $pkg|sed 's|\.\.tar.*|||') # Package directory

tar -xf $file
pushd $packagedir

mkdir build
cd     build

cmake -D CMAKE_INSTALL_PREFIX=$KF6_PREFIX \
      -D CMAKE_INSTALL_LIBEXECDIR=libexec \
      -D CMAKE_BUILD_TYPE=Release          \
      -D BUILD_QT5=OFF                   \
      -D BUILD_TESTING=OFF              \
      -W no-dev .. &&

make
as_root make install
popd

as_root rm -rf $packagedir
as_root /sbin/ldconfig

done < plasma-6.1.4.md5

exit

```

If you did not set `$KF6_PREFIX` to `/usr`, create symlinks to allow display managers to find Plasma, and to allow the XDG Desktop Portal to be detected. As the `root` user:

```

# Setup xsessions (X11 sessions)
install -dvm 755 /usr/share/xsessions
cd /usr/share/xsessions

[ -e plasma.desktop ] ||
ln -sfv $KF6_PREFIX/share/xsessions/plasmamax11.desktop

# Setup wayland-sessions
install -dvm 755 /usr/share/wayland-sessions
cd /usr/share/wayland-sessions

[ -e plasmawayland.desktop ] ||
ln -sfv $KF6_PREFIX/share/wayland-sessions/plasma.desktop

# Setup xdg-desktop-portal
install -dvm 755 /usr/share/xdg-desktop-portal
cd /usr/share/xdg-desktop-portal

[ -e kde-portals.conf ] ||
ln -sfv $KF6_PREFIX/share/xdg-desktop-portal/kde-portals.conf

# Setup kde portal
install -dvm 755 /usr/share/xdg-desktop-portal/portals
cd /usr/share/xdg-desktop-portal/portals

[ -e kde.portal ] ||
ln -sfv $KF6_PREFIX/share/xdg-desktop-portal/portals/kde.portal

```

Configuring Plasma

Linux PAM Configuration

If you built Plasma with the recommended Linux PAM support, create necessary configuration files by running the following commands as the `root` user:

```

cat > /etc/pam.d/kde << "EOF"
# Begin /etc/pam.d/kde

auth    requisite    pam_nologin.so
auth    required     pam_env.so

```

```

auth      required      pam_succeed_if.so uid >= 1000 quiet
auth      include       system-auth

account  include       system-account
password include     system-password
session   include      system-session

# End /etc/pam.d/kde
EOF

cat > /etc/pam.d/kde-np << "EOF"
# Begin /etc/pam.d/kde-np

auth      requisite    pam_nologin.so
auth      required     pam_env.so

auth      required    pam_succeed_if.so uid >= 1000 quiet
auth      required    pam_permit.so

account  include      system-account
password include    system-password
session   include     system-session

# End /etc/pam.d/kde-np
EOF

cat > /etc/pam.d/kscreensaver << "EOF"
# Begin /etc/pam.d/kscreensaver

auth      include system-auth
account  include system-account

# End /etc/pam.d/kscreensaver
EOF

```

Starting Plasma

You can start Plasma from a TTY, using [xinit-1.4.2](#).

To start Plasma using [xinit-1.4.2](#), run the following commands:

```

cat > ~/.xinitrc << "EOF"
dbus-launch --exit-with-x11 $KF6_PREFIX/bin/startplasma-x11
EOF

startx

```

The X session starts on the first unused virtual terminal, normally vt7. You can switch to another vtn simultaneously pressing the keys Ctrl-Alt-Fn ($n=1, 2, \dots$). To switch back to the X session, normally started at vt7, use Ctrl-Alt-F7. The vt where the command `startx` was executed will display many messages, including X starting messages, applications automatically started with the session, and eventually, some warning and error messages. You may prefer to redirect those messages to a log file, which not only will keep the initial vt uncluttered, but can also be used for debugging purposes. This can be done starting X with:

```
startx &> ~/x-session-errors
```

When shutting down or rebooting, the shutdown messages appear on the vt where X was running. If you wish to see those messages, simultaneously press keys Alt-F7 (assuming that X was running on vt7).

Contents

Installed Programs: There are too many plasma programs (over 50 in /opt/kf6/bin) to list separately here.

Installed Libraries: There are too many plasma libraries (over 250 in /opt/kf6/lib) to list separately here.

Installed Directories: There are too many plasma directories (over 2700 in /opt/kf6) to list separately here.

Part VIII. GNOME

Chapter 33. GNOME Libraries and Desktop

The objective of this section is to build a GNOME desktop.

Libraries

Gcr-3.41.2

Introduction to Gcr

The Gcr package contains libraries used for displaying certificates and accessing key stores. It also provides the viewer for crypto files on the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gcr/3.41/gcr-3.41.2.tar.xz>
- Download MD5 sum: 40a754ba44d5e95e4d07656d6302900c
- Download size: 1012 KB
- Estimated disk space required: 33 MB (with tests)
- Estimated build time: 0.2 SBU (with tests; both using parallelism=4)

Gcr Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended), [libgcrypt-1.11.0](#), and [p11-kit-0.25.5](#)

Recommended

[GnuPG-2.4.5](#), [GTK+-3.24.43](#), [libsecret-0.21.4](#), [libxslt-1.1.42](#), and [Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#) and [Valgrind-3.23.0](#)

Installation of Gcr

Install Gcr by running the following commands:

```
sed -i 's:"/desktop:"/org:' schema/*.xml &&
mkdir build &&
cd build &&
meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false   \
            -D ssh_agent=false \
            ..                  &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed -e "/install_dir/s@,\$@ / 'gcr-3.41.2'@" \
-i ../docs/*meson.build
meson configure -D gtk_doc=true
ninja
```

To test the results, issue: `ninja test`. The tests must be run from an X Terminal or similar.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D ssh_agent=false`: Prevent a conflict with [Gcr-4.3.0](#). Omit this switch if you do not plan to install [Gcr-4.3.0](#) and you want to use `gcr-ssh-agent`.

`-D gtk=false`: Use this switch if you haven't installed [GTK+-3.24.43](#). Note that gcr-viewer will not be installed if this is passed to meson.

`-D gtk_doc=true`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

Contents

Installed Program: `gcr-viewer`

Installed Libraries: `libgck-1.so`, `libgcr-base-3.so` and `libgcr-ui-3.so`

Installed Directories: `/usr/include/gck-1`, `/usr/include/gcr-3`, and `/usr/share/gtk-doc/html/{gcr,gck}`

Short Descriptions

`gcr-viewer` is used to view certificates and key files
`libgck-1.so` contains GObject bindings for PKCS#11

Gcr-4.3.0

Introduction to Gcr

The Gcr package contains libraries used for displaying certificates and accessing key stores. It also provides the viewer for crypto files on the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gcr/4.3/gcr-4.3.0.tar.xz>
- Download MD5 sum: 51ae0448c7c42caf0b7deb2639c9e9b8
- Download size: 712 KB
- Estimated disk space required: 30 MB (with tests)
- Estimated build time: 0.1 SBU (with tests; both using parallelism=4)

Gcr Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended), [libgcrypt-1.11.0](#), and [p11-kit-0.25.5](#)

Recommended

[GnuPG-2.4.5](#), [GTK-4.14.5](#), [libsecret-0.21.4](#), [libxslt-1.1.42](#), [OpenSSH-9.8p1](#), and [Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#), [GnuTLS-3.8.7.1](#), and [Valgrind-3.23.0](#)

Installation of Gcr

Note

Both gcr-3 and gcr-4 are coinstallable. This version of the package is used to support GTK-4 applications, such as [gnome-shell-46.4](#) and [Epiphany-46.3](#).

Install Gcr by running the following commands:

```

mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false   \
            ..
            &&
ninja

```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```

sed -e "/install_dir/s@,\$@ / 'gcr-4.3.0'@" \
-i ../docs/*meson.build
meson configure -D gtk_doc=true
ninja

```

To test the results, issue: `ninja test`. The tests must be run from an X Terminal or similar.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk4=false`: Use this switch if you haven't installed [GTK-4.14.5](#). Note that gcr-viewer will not be installed if this is passed to meson.

`-D ssh_agent=false`: Use this switch if you haven't installed [OpenSSH-9.8p1](#), and wish to disable ssh compatibility.

`-D gtk_doc=true`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

`-D crypto=gnutls`: Use this switch if you have [GnuTLS-3.8.7.1](#) installed and wish to use it instead of [libgcrypt-1.11.0](#).

Contents

Installed Program: gcr-viewer-gtk4

Installed Libraries: libgck-2.so, libgcr-4.so

Installed Directories: /usr/include/gck-2, /usr/include/gcr-4, and /usr/share/gtk-doc/html/{gcr,gck}

Short Descriptions

<code>gcr-viewer-gtk4</code>	is used to view certificates and key files
<code>libgck-2.so</code>	contains GObject bindings for PKCS#11
<code>libgcr-4.so</code>	contains functions for accessing key stores and displaying certificates

gsettings-desktop-schemas-46.1

Introduction to GSettings Desktop Schemas

The GSettings Desktop Schemas package contains a collection of GSettings schemas for settings shared by various components of a GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gsettings-desktop-schemas/46/gsettings-desktop-schemas-46.1.tar.xz>
- Download MD5 sum: f1fe79211639016c94df811e8ea6abe6
- Download size: 756 KB
- Estimated disk space required: 15 MB
- Estimated build time: less than 0.1 SBU

GSettings Desktop Schemas Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection)

Installation of GSettings Desktop Schemas

Install GSettings Desktop Schemas by running the following commands:

```
sed -i -r 's:"(/system):"/org/gnome\1:g' schemas/*.in &&
mkdir build &&
cd      build &&
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a “DESTDIR” method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Command Explanations

`sed ... schemas/*.in`: This command fixes some deprecated entries in the schema templates.

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: /usr/include/gsettings-desktop-schemas and /usr/share/GConf/gsettings

libsecret-0.21.4

Introduction to libsecret

The libsecret package contains a GObject based library for accessing the Secret Service API.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libsecret/0.21/libsecret-0.21.4.tar.xz>
- Download MD5 sum: 85f0534c673ccf10daf6fa419728e0d8
- Download size: 204 KB
- Estimated disk space required: 10 MB (with tests)
- Estimated build time: 0.3 SBU (with tests)

libsecret Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended)

Recommended

[libgcrypt-1.11.0](#) (or [GnuTLS-3.8.7.1](#), for cryptography), and [Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#) and [docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [libxslt-1.1.42](#) (to build manual pages), and [Valgrind-3.23.0](#) (can be used in tests)

Optional (Required for the test suite)

[D-Bus Python-1.3.2](#), [Gjs-1.80.2](#), [PyGObject-3.48.2](#) (Python 3 module), and [tpm2-tss](#)

Runtime Dependency

[gnome-keyring-46.2](#)

Note

Any package requiring libsecret expects GNOME Keyring to be present at runtime.

Installation of libsecret

Install libsecret by running the following commands:

```
mkdir bld &&
cd bld &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false   \
            ..
            &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "s/api_version_major/'0.21.4'/"      \
     -i ../docs/reference/libsecret/meson.build &&
meson configure -D gtk_doc=true           &&
ninja
```

Now, as the `root` user:

```
ninja install
```

To test the results, issue: `dbus-run-session ninja test`.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

`-D manpage=false`: Use this switch if you have not installed [libxslt-1.1.42](#) and DocBook packages.

`-D crypto=gnutls`: Use this switch if you want to use [GnuTLS-3.8.7.1](#) for cryptography instead of [libgcrypt-1.11.0](#).

`-D crypto=disabled`: Use this switch if you don't have [GnuTLS-3.8.7.1](#) or [libgcrypt-1.11.0](#) installed. Note that disabling transport encryption support by doing this is not recommended.

Contents

Installed Program: secret-tool

Installed Library: libsecret-1.so

Installed Directories: /usr/include/libsecret-1 and /usr/share/doc/libsecret-0.21.4

Short Descriptions

secret-tool	is a command line tool that can be used to store and retrieve passwords
libsecret-1.so	contains the libsecret API functions

rest-0.9.1

Introduction to rest

The rest package contains a library that was designed to make it easier to access web services that claim to be "RESTful". It includes convenience wrappers for libsoup and libxml to make remote usage of the RESTful API easier.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/rest/0.9/rest-0.9.1.tar.xz>
- Download MD5 sum: b997b83232be3814a1b78530c5700df9
- Download size: 72 KB
- Estimated disk space required: 3.2 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

rest Dependencies

Required

[JSON-GLib-1.8.0](#), [libsoup-3.4.4](#), and [make-ca-1.14](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[Gi-DocGen-2024.1](#), [libadwaita-1.5.3](#) and [gtksourceview5-5.12.1](#) (to build the demo), and [Vala-0.56.17](#)

Note

An Internet connection is needed for some tests of this package.

Installation of rest

Install rest by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D examples=false   \
            -D gtk_doc=false    \
            ..
            &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/output/s/librest-1.0/rest-0.9.1/" -i ../docs/meson.build &&
meson configure -D gtk_doc=true                                &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

`-D examples=false`: Remove this option if [libadwaita-1.5.3](#) and [gtksourceview5-5.12.1](#) are installed and you wish to build the demonstration application provided by this package.

`-D vapi=true`: Use this switch if [Vala-0.56.17](#) is installed and you wish to build the Vala bindings provided by this package.

Contents

Installed Programs: librest-demo (optional)

Installed Libraries: librest-1.0.so and librest-extras-1.0.so

Installed Directories: /usr/include/rest-1.0 and /usr/share/gtk-doc/html/rest-1.0

Short Descriptions

<code>librest-demo</code>	provides an example of how to use the RESTful Web API Query functions
<code>librest-1.0.so</code>	contains the RESTful Web API Query functions
<code>librest-extras-1.0.so</code>	contains extra RESTful Web API Query functions

totem-pl-parser-3.26.6

Introduction to Totem PL Parser

The Totem PL Parser package contains a simple GObject-based library used to parse multiple playlist formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/totem-pl-parser/3.26/totem-pl-parser-3.26.6.tar.xz>
- Download MD5 sum: 69dc2cf0e61e6df71ed45156b24b14da
- Download size: 1.4 MB
- Estimated disk space required: 9.6 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Totem PL Parser Dependencies

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [libarchive-3.7.4](#), and [libgcrypt-1.11.0](#)

Optional

[CMake-3.30.2](#), [GTK-Doc-1.34.0](#), [Gvfs-1.54.2](#) (for some tests), [LCOV](#), and [libquvi >= 0.9.1 and libquvi-scripts](#) - if they are installed, then [lua-socket \(git\)](#) is necessary for the tests

Note

An Internet connection is needed for some tests of this package.

Installation of Totem PL Parser

Install Totem PL Parser by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`. The test named `parser` is known to fail.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Libraries: libtotem-plparser-mini.so and libtotem-plparser.so

Installed Directories: /usr/include/totem-pl-parser and /usr/share/gtk-doc/html/totem-pl-parser

Short Descriptions

libtotem-plparser.so	is the Totem Playlist Parser library
libtotem-plparser-mini.so	is the Totem Playlist Parser library, mini version

VTE-0.76.4

Introduction to VTE

The VTE package contains a virtual terminal widget for GTK applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.gnome.org/GNOME/vte/-/archive/0.76.4/vte-0.76.4.tar.gz>
- Download MD5 sum: 5d1a548ad131664e8a638da24fbfd3e9
- Download size: 696 KB
- Estimated disk space required: 17 MB (with tests)
- Estimated build time: 0.4 SBU (using parallelism=4; with tests)

VTE Dependencies

Required

[GTK+-3.24.43](#), [libxml2-2.13.3](#), and [pcre2-10.44](#)

Recommended

[ICU-75.1](#), [GnuTLS-3.8.7.1](#), [GLib-2.80.4](#) (with GObject Introspection), [GTK-4.14.5](#), and [Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#)

Installation of VTE

Install VTE by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed -e "/docdir =/s@\$@/vte-0.76.4@" \
-i ./doc/reference/meson.build &&
meson configure -D docs=true &&
ninja
```

To test the results, issue `ninja test`.

Now, as the `root` user:

```
ninja install &&
rm -v /etc/profile.d/vte.*
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`rm -v /etc/profile.d/vte.*`: This command removes two files installed in `/etc/profile.d` that have no use on a LFS system.

`-D gnutls=false`: Add this switch if you do not want to enable GnuTLS support.

`-D vapi=false`: Add this switch if you do not want to enable vala bindings.

`-D gtk4=false`: Add this switch if you do not want to build the GTK-4 version of VTE.

Contents

Installed Program: vte-2.91 and vte-2.91-gtk4

Installed Library: libvte-2.91.so and libvte-2.91-gtk4.so

Installed Directories: /usr/include/vte-2.91, /usr/include/vte-2.91-gtk4, and (optional) /usr/share/doc/vte-0.76.4

Short Descriptions

<code>vte-2.91</code>	is a test application for the VTE libraries
<code>vte-2.91-gtk4</code>	is a test application for the GTK-4 version of the VTE libraries
<code>libvte-2.91.so</code>	is a library which implements a terminal emulator widget for GTK+ 3
<code>libvte-2.91-gtk4.so</code>	is a library which implements a terminal emulator widget for GTK-4

yelp-xsl-42.1

Introduction to Yelp XSL

The Yelp XSL package contains XSL stylesheets that are used by the Yelp help browser to format Docbook and Mallard documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/yelp-xsl/42/yelp-xsl-42.1.tar.xz>
- Download MD5 sum: c41858c78e34bb8b68a535657a3e15d9
- Download size: 652 KB
- Estimated disk space required: 10 MB
- Estimated build time: less than 0.1 SBU

Yelp XSL Dependencies

Required

[libxslt-1.1.42](#) and [itstool-2.0.7](#)

Installation of Yelp XSL

Install Yelp XSL by running the following commands:

```
./configure --prefix=/usr
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directory: /usr/share/yelp-xsl

geocode-glib-3.26.4

Introduction to Geocode GLib

The Geocode GLib is a convenience library for the Yahoo! Place Finder APIs. The Place Finder web service allows you to do geocoding (finding longitude and latitude from an address), as well as reverse geocoding (finding an address from coordinates).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/geocode-glib/3.26/geocode-glib-3.26.4.tar.xz>
- Download MD5 sum: 4c0dcdb7ee1222435b20acd3d7b68cd1
- Download size: 76 KB
- Estimated disk space required: 4.6 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Geocode GLib Dependencies

Required

[JSON-GLib-1.8.0](#) and [libsoup-3.4.4](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#)

Installation of Geocode GLib

Install Geocode GLib by running the following commands:

```
mkdir build          &&
cd    build          &&

meson setup --prefix /usr      \
            --buildtype=release \
            -D enable-gtk-doc=false \
            -D soup2=false      \
```

```
ninja
```

```
..
```

```
&&
```

To test the results, issue: `LANG=C ninja test`. One test fails because it needs the `sv_SE.utf8` locale, which is not installed by default in LFS.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D enable-gtk-doc=false`: Allow this package to be built without [GTK-Doc-1.34.0](#). Remove this parameter if [GTK-Doc-1.34.0](#) is installed and you wish to rebuild and install the API documentation.

`-D soup2=false`: This switch forces this package to use libsoup-3 for HTTP requests instead of libsoup-2. The packages in BLFS that use geocode-glib now expect libsoup-3 to be used.

Contents

Installed Programs: None

Installed Library: libgeocode-glib-2.so

Installed Directories: /usr/include/geocode-glib-2.0, /usr/libexec/installed-tests/geocode-glib, /usr/share/icons/gnome/scalable/places, and /usr/share/gtk-doc/html/geocode-glib

Short Descriptions

`libgeocode-glib-2.so` contains the Geocode GLib API functions

Gjs-1.80.2

Introduction to Gjs

Gjs is a set of Javascript bindings for GNOME.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gjs/1.80/gjs-1.80.2.tar.xz>
- Download MD5 sum: ae896ab9a1c5589b86fde94a45a2fcfd9
- Download size: 648 KB
- Estimated disk space required: 253 MB (with tests)
- Estimated build time: 0.2 SBU (with tests; with parallelism=4)

Gjs Dependencies

Required

[Cairo-1.18.0](#), [dbus-1.14.10](#), [GLib-2.80.4](#) (with GObject Introspection), and [SpiderMonkey from Firefox-115.14.0](#)

Recommended (required for GNOME)

[GTK+-3.24.43](#) and [GTK-4.14.5](#)

Optional

[Valgrind-3.23.0](#) (for tests), [DTrace](#), [LCOV](#), [sysprof](#), and [Systemtap](#)

Installation of Gjs

Install Gjs by running the following commands:

```

mkdir gjs-build &&
cd    gjs-build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            --wrap-mode=nofallback \
            ..
            &&
ninja

```

To test the results, issue: `ninja test` in a graphical session.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`--wrap-mode=nofallback`: This switch prevents `meson` from using subproject fallbacks for any dependency declarations in the build files, stopping it downloading any optional dependency which is not installed on the system.

`-D profiler=disabled`: prevents building the profiler backend even if `sysprof` is installed.

Contents

Installed Programs: `gjs` (symlink) and `gjs-console`

Installed Library: `libgjs.so`

Installed Directories: `/usr/include/gjs-1.0`, `/usr/lib/gjs`, `/usr/libexec/installed-tests/gjs`, `/usr/share/gjs-1.0`, and `/usr/share/installed-tests/gjs`

Short Descriptions

<code>gjs-console</code>	contains a console to run JavaScript commands
<code>libgjs.so</code>	contains the GNOME JavaScript bindings

gnome-autoar-0.4.4

Introduction to gnome-autoar

The `gnome-autoar` package provides a framework for automatic archive extraction, compression, and management.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-autoar/0.4/gnome-autoar-0.4.4.tar.xz>
- Download MD5 sum: ac1a423f861ed5765a5d03251c00746d
- Download size: 52 KB
- Estimated disk space required: 2.7 MB
- Estimated build time: 0.1 SBU

gnome-autoar Dependencies

Required

[libarchive-3.7.4](#) and [GTK+-3.24.43](#)

Recommended

[Vala-0.56.17](#)

Optional

Installation of gnome-autoar

Install gnome-autoar by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D vapi=true        \
            -D tests=true       \
            ..
            &&
ninja
```

To test the results, issue: `ninja test`

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Libraries: libgnome-autoar-0.so and libgnome-autoar-gtk-0.so

Installed Directories: /usr/include/gnome-autoar-0 and /usr/share/gtk-doc/html/gnome-autoar

Short Descriptions

<code>libgnome-autoar-0.so</code>	provides API functions for automatic archive management
<code>libgnome-autoar-gtk-0.so</code>	provides GTK+ widgets to aid in automatic archive management

gnome-desktop-44.1

Introduction to GNOME Desktop

The GNOME Desktop package contains a library that provides an API shared by several applications on the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-desktop/44/gnome-desktop-44.1.tar.xz>
- Download MD5 sum: eda77690fcb351558ea0d1716a55e90b
- Download size: 752 KB
- Estimated disk space required: 23 MB
- Estimated build time: 0.2 SBU

GNOME Desktop Dependencies

Required

[gsettings-desktop-schemas-46.1](#), [GTK+-3.24.43](#), [GTK-4.14.5](#), [ISO Codes-4.16.0](#), [itstool-2.0.7](#), [libseccomp-2.5.5](#), [libxml2-2.13.3](#), and [xkeyboard-config-2.42](#)

Recommended

[bubblewrap-0.9.0](#) (needed for thumbnailers in Nautilus) and [GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#)

Installation of GNOME Desktop

Install GNOME Desktop by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            ..
            &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this parameter if you wish to build the API documentation.

`-D installed_tests=true`: Use this parameter you wish to enable the installed tests.

Contents

Installed Programs: None

Installed Library: libgnome-bg-4.so, libgnome-desktop-3.so, libgnome-desktop-4.so, and libgnome-rr-4.so

Installed Directories: /usr/include/gnome-desktop-3.0, /usr/include/gnome-desktop-4.0, /usr/libexec/gnome-desktop-debug, /usr/share/gtk-doc/html/gnome-desktop3 (optional), and /usr/share/help/*/{fdl,gpl,lGPL}

Short Descriptions

`libgnome-desktop-3.so` contains functions shared by several GNOME applications

gnome-menus-3.36.0

Introduction to GNOME Menus

The GNOME Menus package contains an implementation of the draft [Desktop Menu Specification](#) from freedesktop.org. It also contains the GNOME menu layout configuration files and `.directory` files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-menus/3.36/gnome-menus-3.36.0.tar.xz>
- Download MD5 sum: a8fd71fcf31a87fc799d80396a526829
- Download size: 492 KB
- Estimated disk space required: 8.5 MB
- Estimated build time: less than 0.1 SBU

GNOME Menus Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended)

Installation of GNOME Menus

Install GNOME Menus by running the following commands:

```
./configure --prefix=/usr \
--sysconfdir=/etc \
--disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Libraries: libgnome-menu-3.so

Installed Directories: /etc/xdg/menus, /usr/include/gnome-menus-3.0, and /usr/share/desktop-directories

Short Descriptions

libgnome-menu-3.so	contains functions required to support GNOME's implementation of the Desktop Menu Specification
--------------------	---

gnome-online-accounts-3.50.4

Introduction to GNOME Online Accounts

The GNOME Online Accounts package contains a framework used to access the user's online accounts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-online-accounts/3.50/gnome-online-accounts-3.50.4.tar.xz>
- Download MD5 sum: 93cc587f17ad9c14f03bcd0b9c124f8b
- Download size: 400 KB
- Estimated disk space required: 14 MB
- Estimated build time: 0.2 SBU

GNOME Online Accounts Dependencies

Required

[Gcr-4.3.0](#), [JSON-GLib-1.8.0](#), [libadwaita-1.5.3](#), [rest-0.9.1](#), and [Vala-0.56.17](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#), [MIT Kerberos V5-1.21.3](#), and [Valgrind-3.23.0](#)

Installation of GNOME Online Accounts

Note

The Google API Key and OAuth tokens below are specific to LFS. If using these instructions for another distro, or if you intend to distribute binary copies of the software using these instructions, please obtain your own keys following the instructions located at <https://www.chromium.org/developers/how-tos/api-keys>.

Install GNOME Online Accounts by running the following commands:

```
mkdir build &&
cd build &&

meson setup \
    --prefix=/usr \
    --buildtype=release \
    -D kerberos=false \
    -D google_client_secret=5ntt6GbbkjnTVXx-MSxbmx5e \
    -D google_client_id=595013732528-1lk8trb03f01dpqq6nprjp1s79596646.apps.googleusercontent.com \
    ... &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D google_*`: These switches set the BLFS OAuth ID and secret for Google services.

`-D gtk_doc=true`: Use this switch if you have [GTK-Doc-1.34.0](#) installed and you wish to generate the API documentation.

`-D kerberos=true`: Use this switch if you have installed [MIT Kerberos V5-1.21.3](#) and wish to use it with GNOME Online Accounts.

Contents

Installed Programs: goa-daemon and goa-identity-service (library executables)

Installed Libraries: libgoa-1.0.so and libgoa-backend-1.0.so

Installed Directories: /usr/include/goa-1.0, /usr/lib/goa-1.0, and /usr/share/gtk-doc/html/goa (optional)

Short Descriptions

<code>goa-daemon</code>	is the GNOME Online Accounts Daemon
<code>libgoa-1.0.so</code>	contains the GNOME Online Accounts API functions
<code>libgoa-backend-1.0.so</code>	contains functions used by GNOME Online Accounts backends

Grilo-0.3.16

Introduction to Grilo

Grilo is a framework focused on making media discovery and browsing easy for applications and application developers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/grilo/0.3/grilo-0.3.16.tar.xz>
- Download MD5 sum: e81c4d7e182eb6448b9f0458f52511a3
- Download size: 236 KB
- Estimated disk space required: 13 MB (with tests)

- Estimated build time: 0.2 SBU (with tests)

Grilo Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended) and [libxml2-2.13.3](#)

Recommended

[GTK+-3.24.43](#), [libsoup-3.4.4](#), [totem-pl-parser-3.26.6](#), and [Vala-0.56.17](#)

Optional

[liboauth-1.0.3](#) and [GTK-Doc-1.34.0](#)

Installation of Grilo

Install Grilo by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D enable-gtk-doc=false \
            ..
            &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D enable-gtk-doc=false`: This option disables generating documentation. If you have [GTK-Doc-1.34.0](#) installed and wish to generate documentation, remove this option.

Contents

Installed Programs: grilo-test-ui-0.3, grl-inspect-0.3 and grl-launch-0.3

Installed Libraries: libgrilo-0.3.so, libgrlnet-0.3.so and libgrlpls-0.3.so

Installed Directories: /usr/include/grilo-0.3

Short Descriptions

<code>grilo-test-ui-0.3</code>	is a simple playground application that you can use to test the framework and its plugins
<code>grl-inspect-0.3</code>	is a tool that prints out information on available Grilo sources
<code>grl-launch-0.3</code>	is a tool to run Grilo operations from command line
<code>libgrilo.so</code>	provides the Grilo framework
<code>libgrlnet.so</code>	provides Grilo networking helpers for plug-ins
<code>libgrlpls.so</code>	provides playlist handling functions

libgdata-0.18.1

Introduction to libgdata

The libgdata package is a GLib-based library for accessing online service APIs using the GData protocol, most notably, Google's services. It provides APIs to access the common Google services and has full asynchronous support.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgdata/0.18/libgdata-0.18.1.tar.xz>
- Download MD5 sum: 92b058d1a0af5d1b96c86c21820f1eff
- Download size: 832 KB
- Estimated disk space required: 54 MB (with tests)
- Estimated build time: 0.5 SBU (with tests)

libgdata Dependencies

Required

[libsoup-2.74.3](#), [gnome-online-accounts-3.50.4](#), [GTK+-3.24.43](#), [JSON-GLib-1.8.0](#), and [Vala-0.56.17](#)

Recommended

[Gcr-3.41.2](#) and [GLib-2.80.4](#) (with GObject Introspection)

Optional

[GTK-Doc-1.34.0](#) (for documentation), and [liboauth-1.0.3](#) (for OAuth v1 support)

Installation of libgdata

Install libgdata by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D gtk_doc=false \
            -D always_build_tests=false \
            ..
            &&
ninja
```

This package does not come with a functional test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: Remove this if you have [GTK-Doc-1.34.0](#) installed and want to rebuild the documentation with it.

`-D oauth1(enabled)`: Use this option if you wish to build in support for OAuth v1. Note that you must have [liboauth-1.0.3](#) installed in order to enable this option.

Contents

Installed Programs: None

Installed Libraries: libgdata.so

Installed Directories: /usr/include/libgdata and /usr/share/gtk-doc/html/gdata

Short Descriptions

libgdata.so contains the libgdata API functions

Introduction to libgee

The libgee package is a collection library providing GObject based interfaces and classes for commonly used data structures.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgee/0.20/libgee-0.20.6.tar.xz>
- Download MD5 sum: 8b9001f47e15ef7a1776ac1f5bb015a0
- Download size: 676 KB
- Estimated disk space required: 38 MB (with tests)
- Estimated build time: 0.5 SBU (with tests)

libgee Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended) and [Vala-0.56.17](#)

Recommended

[Vala-0.56.17](#)

Optional

[LCOV](#)

Installation of libgee

First, force the Vala code to be regenerated so that it is compatible with gcc-14:

```
find . -name \*.vala -exec touch {} \;
```

Install libgee by running the following commands:

```
./configure --prefix=/usr --enable-vala &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libgee-0.8.so

Installed Directory: /usr/include/gee-0.8

Short Descriptions

libgee-0.8.so contains the libgee API functions

libgtop-2.41.3

Introduction to libgtop

The libgtop package contains the GNOME top libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgtop/2.41/libgtop-2.41.3.tar.xz>
- Download MD5 sum: 465db9f4f695c298d9c48dcf7f32a9c0
- Download size: 740 KB
- Estimated disk space required: 19 MB
- Estimated build time: 0.2 SBU

libgtop Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection recommended) and [Xorg Libraries](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of libgtop

Install libgtop by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: libgtop_daemon2 and libgtop_server2

Installed Library: libgtop-2.0.so

Installed Directories: /usr/include/libgtop-2.0 and /usr/share/gtk-doc/html/libgtop

Short Descriptions

libgtop-2.0.so	contains functions that allow access to system performance data
----------------	---

libgweather-4.4.2

Introduction to libgweather

The libgweather package is a library used to access weather information from online services for numerous locations.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libgweather/4.4/libgweather-4.4.2.tar.xz>
- Download MD5 sum: 57ec40f5ac366b7d9757580913bf2e3b
- Download size: 2.7 MB
- Estimated disk space required: 98 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libgweather Dependencies

Required

[geocode-glib-3.26.4](#), [GTK+-3.24.43](#), [libsoup-3.4.4](#), and [PyGObject-3.48.2](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [libxml2-2.13.3](#), and [Vala-0.56.17](#)

Optional

[Gi-DocGen-2024.1](#) (gi-docgen is also provided as a meson subproject, which will be used if `-D gtk_doc=false` is not passed to `meson`), [LLVM-18.1.7](#) (for clang-format), and [pylint](#)

Note

An Internet connection is needed for some tests of this package.

Installation of libgweather

Install libgweather by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D gtk_doc=false   \
            ..                  &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "s/libgweather_full_version/'libgweather-4.4.2'/" \
     -i ../doc/meson.build          &&
meson configure -D gtk_doc=true    &&
ninja
```

One test needs that the locale files be installed on the system, so it is better to run the tests after installing the package.

Now, as the `root` user:

```
ninja install
```

To test the results, issue: `LC_ALL=C ninja test`.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=false`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

Contents

Installed Programs: None

Installed Libraries: libgweather-4.so

Installed Directories: /usr/lib/libgweather-4, /usr/include/libgweather-4.0, /usr/share/gtk-doc/html/libgweather-4.0 (optional), and /usr/share/libgweather-4

Short Descriptions

libgweather-4.so contains functions that allow for the retrieval of weather information

libpeas-1.36.0

Introduction to libpeas

libpeas is a GObject based plugins engine, and is targeted at giving every application the chance to assume its own extensibility.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libpeas/1.36/libpeas-1.36.0.tar.xz>
- Download MD5 sum: b3dd31a79c47af0cbf22f2d6bf52bc7d
- Download size: 192 KB
- Estimated disk space required: 10 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

libpeas Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection) and [GTK+-3.24.43](#)

Recommended

[libxml2-2.13.3](#) and [PyGObject-3.48.2](#)

Optional

[Gi-DocGen-2024.1](#), [Glade](#), [embed](#), [LGI](#) (for LUA bindings, built with LUA-5.1), with either [luajit](#) or [LUA-5.1](#)

Installation of libpeas

Install libpeas by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr \
            --buildtype=release \
            --wrap-mode=nofallback \
            ...
            &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/docs_dir =/s@\$@/ 'libpeas-1.36.0'@" \
     -i ../docs/reference/meson.build &&
meson configure -D gtk_doc=true &&
ninja
```

To test the results, issue: `ninja test`. An active graphical session with bus address is necessary to run the tests.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`--wrap-mode=nofallback`: This switch prevents `meson` from using subproject fallbacks for any dependency declarations in the build files, stopping it downloading any optional dependency which is not installed on the system.

`-D vapi=true`: Add this switch if you wish to generate vapi (vala) data.

`-D demos=false`: Add this switch if you do not wish to build the demo programs.

Contents

Installed Program: peas-demo

Installed Libraries: libpeas-1.0.so and libpeas-gtk-1.0.so

Installed Directories: /usr/include/libpeas-1.0, /usr/lib/libpeas-1.0, /usr/lib/peas-demo and /usr/share/gtk-doc/html/libpeas (optional)

Short Descriptions

<code>peas-demo</code>	is the Peas demo program
<code>libpeas-1.0.so</code>	contains the libpeas API functions
<code>libpeas-gtk-1.0.so</code>	contains the libpeas GTK+ widgets

libshumate-1.2.3

Introduction to libshumate

The libshumate package contains a GTK-4 widget to display maps.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libshumate/1.2/libshumate-1.2.3.tar.xz>
- Download MD5 sum: 4b8919088922eac80b103b464d7c5b8b
- Download size: 316 KB
- Estimated disk space required: 9.1 MB
- Estimated build time: 0.2 SBU

libshumate Dependencies

Required

[GTK-4.14.5](#), [libsoup-3.4.4](#), and [Protobuf-c-1.5.0](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) (required for gnome-maps)

Optional

[Gi-DocGen-2024.1](#), [Valgrind-3.23.0](#), and [sysprof](#)

Installation of libshumate

Install libshumate by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            --wrap-mode=nodownload \
            -D gtk_doc=false     \
            ..                  &&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed -e 's/lib_version/version/' \
     -i ../docs/meson.build &&
```

```
meson configure -D gtk_doc=true &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`--wrap-mode=nodownload`: This switch prevents `meson` from downloading any optional dependency which is not installed on the system.

Contents

Installed Programs: None

Installed Libraries: libshumate-1.0.so

Installed Directories: /usr/include/shumate-1.0 and /usr/share/doc/libshumate-1.2.3 (optional)

Short Descriptions

libshumate-1.0.so contains functions that provide a GTK-4 widget to display maps

evolution-data-server-3.52.4

Introduction to Evolution Data Server

The Evolution Data Server package provides a unified backend for programs that work with contacts, tasks, and calendar information. It was originally developed for Evolution (hence the name), but is now used by other packages as well.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/evolution-data-server/3.52/evolution-data-server-3.52.4.tar.xz>
- Download MD5 sum: a7a269c877206f36c72f2c3e37f79e9c
- Download size: 4.8 MB
- Estimated disk space required: 181 MB (with tests)
- Estimated build time: 0.7 SBU (using parallelism=4)

Evolution Data Server Dependencies

Required

[libical-3.0.18](#), [libsecret-0.21.4](#), [NSS-3.103](#), and [SQLite-3.46.1](#)

Recommended

[gnome-online-accounts-3.50.4](#), [GLib-2.80.4](#) (with GObject Introspection), [GTK+-3.24.43](#), [GTK-4.14.5](#), [ICU-75.1](#), [libcanberra-0.30](#), [libgweather-4.4.2](#), [Vala-0.56.17](#), and [WebKitGTK-2.44.3](#)

Optional

[GTK-Doc-1.34.0](#), [MIT Kerberos V5-1.21.3](#), a [MTA](#) (that provides a `sendmail` command), [OpenLDAP-2.6.8](#), [Berkeley DB](#) (deprecated), and [libphonenumbers](#)

Installation of Evolution Data Server

Install Evolution Data Server by running the following commands:

```

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D SYSCONF_INSTALL_DIR=/etc \
      -D ENABLE_VALA_BINDINGS=ON \
      -D ENABLE_INSTALLED_TESTS=ON \
      -D WITH_OPENLDAP=OFF \
      -D WITH_KRB5=OFF \
      -D ENABLE_INTROSPECTION=ON \
      -D ENABLE_GTK_DOC=OFF \
      -D WITH_LIBDB=OFF \
      -W no-dev -G Ninja .. &&
ninja

```

Now, as the `root` user:

```
ninja install
```

To test the results, issue: `ninja test`.

Command Explanations

`-D ENABLE_VALA_BINDINGS=ON`: This switch enables building the Vala bindings. Remove it if you don't have [Vala-0.56.17](#) installed.

`-D ENABLE_GTK_DOC=OFF`: This switch disables building the API documentation. It is broken for this package due to the use of a long deprecated gtk-doc program that is no longer available.

`-D WITH_LIBDB=OFF`: This switch allows building this package without [Berkeley DB](#) (deprecated). [SQLite-3.46.1](#) is used for normal operation.

`-D ENABLE_OAUTH2_WEBKITGTK4=OFF`: Use this switch if you did not build [WebKitGTK-2.44.3](#) with GTK-4.

Note

To enable many of the optional dependencies, review the information from `cmake -L CMakeLists.txt` for the necessary parameters you must pass to the `cmake` command.

Contents

Installed Programs: None

Installed Libraries: libcamel-1.2.so, libbackend-1.2.so, libebook-1.2.so, libebook-contacts-1.2.so, libecal-2.0.so, libedata-book-1.2.so, libedata-cal-2.0.so, libedataserver-1.2.so, libedataserverui-1.2.so, libedataserverui4-1.0.so, and libetestserverutils.so

Installed Directories: /usr/include/evolution-data-server, /usr/lib{,exec}/evolution-data-server, /usr/share/evolution-data-server, /usr/share/installed-tests/evolution-data-server, and /usr/share/pixmaps/evolution-data-server

Short Descriptions

libcamel-1.2.so	is the Evolution MIME message handling library
libbackend-1.2.so	is the utility library for Evolution Data Server Backends
libebook-1.2.so	is the client library for Evolution address books
libebook-contacts-1.2.so	is the client library for Evolution contacts
libecal-1.2.so	is the client library for Evolution calendars
libedata-book-1.2.so	is the backend library for Evolution address books
libedata-cal-1.2.so	is the backend library for Evolution calendars
libedataserver-1.2.so	is the utility library for Evolution Data Server
libedataserverui-3.0.so	is the GUI utility library for Evolution Data Server
libedataserverui4-1.0.so	is the GTK-4 based GUI utility library for Evolution Data Server
libetestserverutils.so	is the server test utility library for Evolution Data Server

Tracker-3.7.3

Introduction to Tracker

Tracker is the file indexing and search provider used in the GNOME desktop environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/tracker/3.7/tracker-3.7.3.tar.xz>
- Download MD5 sum: 65cd2945506b7303e9eea493d56431d8
- Download size: 1.7 MB
- Estimated disk space required: 58 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

Tracker Dependencies

Required

[JSON-GLib-1.8.0](#), [libseccomp-2.5.5](#), and [Vala-0.56.17](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [ICU-75.1](#), [libsoup-3.4.4](#), [PyGObject-3.48.2](#), [SQLite-3.46.1](#), and [tracker-miners-3.7.3](#) (runtime)

Optional

[asciidoc-10.2.1](#), [Avahi-0.8](#), [Graphviz-12.1.0](#), [libsoup-2.74.3](#), [bash-completion](#), and [libstemmer](#)

Installation of Tracker

Fix the location to install the documentation into:

```
mv docs/reference/libtracker-sparql/doc/{Tracker-3.0,tracker-3.7.3} &&
sed '/docs_name/s/Tracker-3.0/tracker-3.7.3/' \
-i docs/reference/libtracker-sparql/meson.build
```

Install Tracker by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D man=false        \
            ..                  &&
ninja
```

Now, as the `root` user:

```
ninja install
```

To test the results, issue: `meson configure -D debug=true && ninja test`. The test suite should be run from a graphical session.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D man=false`: This switch prevents the build process from generating man pages. Omit this switch if you have [asciidoc-10.2.1](#) installed and wish to generate and install the man pages.

`meson configure -D debug=true`: This command enables some debug checks necessary for the test suite. We don't want to enable them for the installed Tracker 3 libraries and programs, so we run the test suite after installation.

Contents

Installed Programs: tracker3, tracker3-endpoint, tracker3-export, tracker3-help, tracker3-import, tracker3-sparql, and tracker3-sql

Installed Library: libtracker-sparql-3.0.so

Installed Directories: /usr/{include,lib}/tracker-3.0, /usr/libexec/tracker3, /usr/share/tracker3, and /usr/share/doc/tracker-3.7.3 (optional)

Short Descriptions

<code>tracker3</code>	is a control program for the indexer
<code>tracker3-endpoint</code>	creates a SPARQL endpoint
<code>tracker3-export</code>	exports all data from a Tracker database
<code>tracker3-help</code>	shows manual pages related to Tracker
<code>tracker3-import</code>	imports data into a Tracker database
<code>tracker3-sparql</code>	uses SparQL to query a Tracker database
<code>tracker3-sql</code>	uses SQL to query a Tracker database
<code>libtracker-sparql-3.0.so</code>	contains resource management and database functions

Tracker-miners-3.7.3

Introduction to Tracker-miners

The Tracker-miners package contains a set of data extractors for Tracker.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/tracker-miners/3.7/tracker-miners-3.7.3.tar.xz>
- Download MD5 sum: 8c1b48f2fa57d888967de9b5ff348fc1
- Download size: 4.3 MB
- Estimated disk space required: 36 MB (with tests)
- Estimated build time: 0.3 SBU (with parallelism=4; add as much as 1.0 SBU for tests, dependent on disk speed)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/tracker-miners-3.7.3-upstream_fixes-1.patch

Tracker-miners Dependencies

Required

[gst-plugins-base-1.24.7](#), [Tracker-3.7.3](#), [Exempi-2.6.5](#), and [gexiv2-0.14.3](#)

Recommended

[glib-5.2.2](#), [gst-plugins-good-1.24.7](#) (runtime), [gst-libav-1.24.7](#) (runtime), [ICU-75.1](#), [libexif-0.6.24](#), [libgxpath-0.3.2](#), [libseccomp-2.5.5](#), [Poppler-24.08.0](#), and [UPower-1.90.4](#)

Optional

[asciidoc-10.2.1](#), [CMake-3.30.2](#), [DConf-0.40.0](#), [FFmpeg-7.0.2](#), [libgsf-1.14.52](#), [NetworkManager-1.48.8](#), [taglib-2.0.1](#), [totem-pl-parser-3.26.6](#), [libcue](#), [libgrss](#), [libitpcdata](#), [libosinfo](#), and [gupnp](#)

Kernel Configuration

Enable the following options in the kernel configuration, then recompile the kernel and reboot if necessary:

```

Security options --->
[*]   Enable different security models [SECURITY]
[*]   Landlock support [SECURITY_LANDLOCK]
# List more Linux Security Modules here (separated with comma) if needed,
# for example 'landlock,lockdown,smack':
(landlock) Ordered list of enabled LSMs [LSM]

```

Installation of Tracker-miners

Note

If you plan to run the tests, some timeouts are too short when using spinning disks. There are two places where timeouts are used: first, individual tests inside a group of tests have a default timeout of 10s. This can be changed by setting the environment variable `TRACKER_TESTS_AWAIT_TIMEOUT` to the desired value when running the tests (see below). Second, a global timeout for a group of tests is fixed at configuration time. The default value in the `functional-tests` directory (other directories have only short lived tests) may be increased with the following command (replace 200 with a value suitable for your machine):

```
sed -i s/120/200/ tests/functional-tests/meson.build
```

First, fix one issue in the TIFF extractor and another issue in the Landlock sandbox:

```
patch -Np1 -i ../tracker-miners-3.7.3-upstream_fixes-1.patch
```

Install Tracker-miners by running the following commands:

```

mkdir build &&
cd      build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D man=false        \
            -D miner_rss=false \
            ..
            &&
ninja

```

Now, as the `root` user:

```
ninja install
```

To test the results, issue (adjust the individual test timeout to a value suitable for your machine, see the note above):

```

meson configure -D debug=true &&
dbus-run-session env LC_ALL=C.UTF-8 TRACKER_TESTS_AWAIT_TIMEOUT=20 \
                  ninja test &&
rm -rf ~/tracker-tests

```

Note

The tests create files in the user directory (up to 24 MB), so they have to be removed afterwards.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D man=false`: This switch prevents the build process from generating man pages. Omit this switch if you have [asciidoc-10.2.1](#) installed and wish to generate and install the man pages.

`-D miner_rss=false`: This switch prevents using libgrss for indexing RSS feeds because libgrss is insecure and no longer in BLFS.

`-D seccomp=false`: This option disables the seccomp system call filter. On some architectures, such as i686 and ARM, the functions that tracker-miners uses are not guarded properly, and tracker-miners will get killed with a SIGSYS as a result. Note that disabling seccomp may cause the system to be compromised worse in the event that a security vulnerability in tracker-miners or its dependencies is exploited.

`-D landlock=disabled`: This switch disables the Landlock file access sandbox. Use it if you don't want to build the kernel with Landlock support. Note that disabling Landlock may cause the system to be compromised worse in the event that a security vulnerability in tracker-miners or its dependencies is exploited.

`-D battery_detection=none`: Use this option if you have not installed the recommended dependency upower. Do not use this option if your system has a battery (laptop battery or UPS), or tracker-miners may waste the battery power and lifespan when the A/C power is unavailable.

Contents

Installed Programs: tracker3-daemon, tracker3-extract, tracker3-index, tracker3-info, tracker3-reset, tracker3-search, tracker3-status, and tracker3-tag

Installed Libraries: Several modules under /usr/lib/tracker-miners-3.0

Installed Directories: /usr/lib/tracker-miners-3.0, /usr/libexec/tracker3, and /usr/share/tracker3-miners

Short Descriptions

<code>tracker3-daemon</code>	starts, stops, restarts, and lists daemons responsible for indexing content
<code>tracker3-extract</code>	extracts metadata from a file
<code>tracker3-index</code>	indexes content using the Tracker filesystem miner
<code>tracker3-info</code>	retrieves all information available for a certain file
<code>tracker3-reset</code>	resets the Tracker index and configuration
<code>tracker3-search</code>	searches for content by type or across all types
<code>tracker3-status</code>	provides status and statistics on data that has been indexed
<code>tracker3-tag</code>	adds, removes, and lists tags

GSound-1.0.3

Introduction to GSound

The gsound package contains a small library for playing system sounds.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gsound/1.0/gsound-1.0.3.tar.xz>
- Download MD5 sum: 7338c295034432a6e782fd20b3d04b68
- Download size: 24 KB
- Estimated disk space required: 864 KB
- Estimated build time: less than 0.1 SBU

gsound Dependencies

Required

[libcanberra-0.30](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#) (for generating documentation)

Installation of GSound

Install gsound by running the following commands:

```
mkdir build &&
cd      build &&
```

```
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: gsound-play

Installed Libraries: libgsound.so

Installed Directories: /usr/share/gtk-doc/html/gsound

Short Descriptions

<code>gsound-play</code>	plays system sounds through the libgsound interface
<code>libgsound.so</code>	contains API functions for playing system sounds

xdg-desktop-portal-gnome-46.2

Introduction to xdg-desktop-portal-gnome

`xdg-desktop-portal-gnome` is a backend for `xdg-desktop-portal`, that is using GTK and various pieces of GNOME infrastructure.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/xdg-desktop-portal-gnome/46/xdg-desktop-portal-gnome-46.2.tar.xz>
- Download MD5 sum: a57a52d9488463277445bac40507487e
- Download size: 160 KB
- Estimated disk space required: 10 MB
- Estimated build time: 0.2 SBU

xdg-desktop-portal-gnome Dependencies

Required

[gnome-backgrounds-46.0](#), [gnome-desktop-44.1](#), [GTK-4.14.5](#), [libadwaita-1.5.3](#), [xdg-desktop-portal-1.18.2](#), and [xdg-desktop-portal-gtk-1.15.1](#) (at runtime)

Installation of xdg-desktop-portal-gnome

Install `xdg-desktop-portal-gnome` by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a “DESTDIR” method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Contents

Installed Program: one daemon in `/usr/libexec`

Installed Library: None

Installed Directory: `/usr/share/xdg-desktop-portal` (if no other `xdg-desktop-portal` backend is installed)

GNOME Desktop Components

DConf-0.40.0 / DConf-Editor-45.0.1

Introduction to DConf

The DConf package contains a low-level configuration system. Its main purpose is to provide a backend to GSettings on platforms that don't already have configuration storage systems.

The DConf-Editor, as the name suggests, is a graphical editor for the DConf database. Installation is optional, because `gsettings` from [GLib-2.80.4](#) provides similar functionality on the commandline.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/dconf/0.40/dconf-0.40.0.tar.xz>
- Download MD5 sum: ac8db20b0d6b996d4bbbeb96463d01f0
- Download size: 115 KB
- Estimated disk space required: 7.0 MB (with tests)
- Estimated build time: 0.1 SBU (with tests)

Additional Downloads

- Download (HTTP): <https://download.gnome.org/sources/dconf-editor/45/dconf-editor-45.0.1.tar.xz>
- Download MD5 sum: 82b2f5d396e95757ad7eaf89c82decd6
- Download size: 596 KB
- Estimated disk space required: 21 MB
- Estimated build time: 0.3 SBU

DConf Dependencies

Required

[dbus-1.14.10](#), [GLib-2.80.4](#), [GTK+-3.24.43](#) (for the editor), [libhandy-1.8.3](#) (for the editor), and [libxml2-2.13.3](#) (for the editor)

Recommended

[libxslt-1.1.42](#) and [Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#) and [bash-completion](#)

Installation of DConf

Install DConf by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr \
            --buildtype=release \
            -D bash_completion=false \
            ..
            &&
ninja
```

To test the results, issue: `ninja test`

As the `root` user:

```
ninja install
```

Now, optionally install the editor:

```
cd ..          &&
tar -xf ../dconf-editor-45.0.1.tar.xz &&
cd dconf-editor-45.0.1           &&

mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

As the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: `dconf` and `dconf-editor`

Installed Libraries: `libdconf.so` and `libdconfsettings.so` (GIO Module installed in `/usr/lib/gio/modules`)

Installed Directories: `/usr/{include,share/gtk-doc/html}/dconf`

Short Descriptions

<code>dconf</code>	is a simple tool for manipulating the DConf database
<code>dconf-editor</code>	is a graphical program for editing the DConf database
<code>libdconf.so</code>	contains the DConf client API functions

gnome-backgrounds-46.0

Introduction to GNOME Backgrounds

The GNOME Backgrounds package contains a collection of graphics files which can be used as backgrounds in the GNOME Desktop environment. Additionally, the package creates the proper framework and directory structure so that you can add your own files to the collection.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-backgrounds/46/gnome-backgrounds-46.0.tar.xz>
- Download MD5 sum: 85907535d2ed9d4cc6158bee41c35110
- Download size: 18 MB
- Estimated disk space required: 44 MB
- Estimated build time: less than 0.1 SBU

***GNO*ME Backgrounds Dependencies**

Required at runtime

[libjxl-0.10.3](#)

Installation of GNOME Backgrounds

Install GNOME Backgrounds by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr ..
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/backgrounds/gnome and /usr/share/gnome-background-properties

Short Descriptions

GNOME Backgrounds are background images for the GNOME Desktop

Gvfs-1.54.2

Introduction to Gvfs

The Gvfs package is a userspace virtual filesystem designed to work with the I/O abstractions of GLib's GIO library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gvfs/1.54/gvfs-1.54.2.tar.xz>
- Download MD5 sum: 587c5b279ec3020c597f3ab3f6a73bbd
- Download size: 1.2 MB
- Estimated disk space required: 30 MB
- Estimated build time: 0.2 SBU (using parallelism=4)

Gvfs Dependencies

Required

[dbus-1.14.10](#), [GLib-2.80.4](#), [Gcr-4.3.0](#), [libusb-1.0.27](#), and [libsecret-0.21.4](#)

Recommended

[GTK+-3.24.43](#), [libcdio-2.1.0](#), [libgudev-238](#), [libsoup-3.4.4](#), [Systemd-256.4](#) (runtime), and [UDisks-2.10.1](#)

Optional

[Apache-2.4.62](#), [Avahi-0.8](#), [BlueZ-5.77](#), [Fuse-3.16.2](#), [gnome-online-accounts-3.50.4](#), [GTK-Doc-1.34.0](#), [libarchive-3.7.4](#), [libgcrypt-1.11.0](#), [libgdata-0.18.1](#), [libxml2-2.13.3](#), [libxslt-1.1.42](#), [OpenSSH-9.8p1](#), [Samba-4.20.4](#), [gnome-desktop-testing](#) (for tests), [libbluray](#), [libgphoto2](#), [libimobiledevice](#), [libmsgraph](#), [libmtp](#), [libnfs](#), and [Twisted](#)

Installation of Gvfs

Install Gvfs by running the following commands:

```
mkdir build &&
cd      build &&

meson setup      \
    --prefix=/usr \
    --buildtype=release \
    -D onedrive=false \
    -D fuse=false \
    -D gphoto2=false \
    -D afc=false \
    -D bluray=false \
    -D nfs=false \
    -D mtp=false \
    -D smb=false \
    -D dnssd=false \
    -D goa=false \
    -D google=false \
    ... &&
ninja
```

The test suite requires gnome-desktop-testing, which is beyond the scope of BLFS.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a "DESTDIR" method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D <option>=false`: These switches are required if the corresponding dependency is not installed. Remove those where you have installed the corresponding application and wish to use it with Gvfs. The dnssd option requires avahi and both goa and google require GNOME Online Accounts. The google option also requires libgdata. The onedrive support requires the libmsgraph package.

`-D cdda=false`: This switch is required if libcdio is not installed. The cdda backend is useless on machines without a CDROM/DVD drive.

Contents

Installed Programs: None

Installed Library: libgvfscommon.so, libgvfsdaemon.so and some under /usr/lib/gio/modules/

Installed Directories: /usr/include/gvfs-client and /usr/{lib,share}/gvfs

Short Descriptions

`libgvfscommon.so` contains the common API functions used in Gvfs programs

gexiv2-0.14.3

Introduction to gexiv2

gexiv2 is a GObject-based wrapper around the Exiv2 library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gexiv2/0.14/gexiv2-0.14.3.tar.xz>
- Download MD5 sum: d4ca9614baa0bc30d142bcad65f09910
- Download size: 384 KB
- Estimated disk space required: 3.3 MB (with tests)
- Estimated build time: 0.1 SBU (with parallelism=4 and tests)

gexiv2 Dependencies

Required

[Exiv2-0.28.3](#)

Recommended

[Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#) (for documentation)

Installation of gexiv2

Install gexiv2 by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, run:

```
meson configure -D tests=true &&
ninja test
```

As the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Libraries: libgexiv2.so

Installed Directories: /usr/include/gexiv2

Short Descriptions

libgexiv2.so provides a wrapper around the Exiv2 library

Nautilus-46.2

Introduction to Nautilus

The Nautilus package contains the GNOME file manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/nautilus/46/nautilus-46.2.tar.xz>
- Download MD5 sum: 88acd6bd9c5d01fb02c8827222c07bc5
- Download size: 3.1 MB
- Estimated disk space required: 88 MB (with tests)
- Estimated build time: 0.5 SBU (with tests, both using parallelism=4)

Nautilus Dependencies

Required

[bubblewrap-0.9.0](#), [gexiv2-0.14.3](#), [gnome-autoar-0.4.4](#), [gnome-desktop-44.1](#), [libadwaita-1.5.3](#), [libnotify-0.8.3](#), [libportal-0.7.1](#), [libseccomp-2.5.5](#), and [Tracker-3.7.3](#)

Recommended

[desktop-file-utils-0.27](#), [Exempi-2.6.5](#), [GLib-2.80.4](#) (with GObject Introspection), [gst-plugins-base-1.24.7](#), [libcloudproviders-0.3.6](#), and [libexif-0.6.24](#)

Optional

[Gi-DocGen-2024.1](#)

Recommended (Runtime)

[adwaita-icon-theme-46.2](#) and [Gvfs-1.54.2](#) (For hotplugging and device mounting to work)

Installation of Nautilus

Fix the location to install the API documentation:

```
sed "/docdir =/s@\$@ / 'nautilus-46.2'@" -i meson.build
```

Install Nautilus by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D selinux=false   \
            -D packagekit=false \
            ..                  &&

ninja
```

To test the results, issue: `ninja test`. The tests need to be run in a graphical environment. One test is known to fail if [tracker-miners-3.7.3](#) is not installed. One test is also known to timeout if the user running the tests has a large home directory.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a "DESTDIR" method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D selinux=false`: This switch disables the use of selinux which isn't supported by BLFS.

`-D packagekit=false`: This switch disables the use of PackageKit which isn't suitable for BLFS.

`-D cloudproviders=false`: Use this switch if you do not have [libcloudproviders-0.3.6](#) installed.

Contents

Installed Programs: nautilus and nautilus-autorun-software

Installed Library: libnautilus-extension.so

Installed Directories: /usr/{include,lib,share}/nautilus and /usr/share/gtk-doc/html/libnautilus-extension (optional)

Short Descriptions

<code>nautilus</code>	is the GNOME file manager
<code>libnautilus-extension.so</code>	supplies the functions needed by the file manager extensions

gnome-bluetooth-46.1

Introduction to GNOME Bluetooth

The GNOME Bluetooth package contains tools for managing and manipulating Bluetooth devices using the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-bluetooth/46/gnome-bluetooth-46.1.tar.xz>
- Download MD5 sum: 6f1f8e6b51c4903727ef41ec6c398f13
- Download size: 300 KB
- Estimated disk space required: 12 MB (with tests)
- Estimated build time: 0.1 SBU (Using parallelism=4, with tests)

GNOME Bluetooth Dependencies

Required

[GTK-4.14.5](#), [gsound-1.0.3](#), [libnotify-0.8.3](#), and [UPower-1.90.4](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [libadwaita-1.5.3](#)

Optional

[GTK-Doc-1.34.0](#) and [dbusmock-0.32.1](#)

Runtime Dependencies

[BlueZ-5.77](#)

Installation of GNOME Bluetooth

Install GNOME Bluetooth by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: `bluetooth-sendto`

Installed Libraries: `libgnome-bluetooth-3.0.so` and `libgnome-bluetooth-ui-3.0.so`

Installed Directories: `/usr/include/gnome-bluetooth-3.0` and `/usr/share/gnome-bluetooth-3.0`

Short Descriptions

`bluetooth-sendto` is a GTK+ application for transferring files over Bluetooth
`libgnome-bluetooth-3.0.so` contains the GNOME Bluetooth API functions

gnome-keyring-46.2

Introduction to GNOME Keyring

The GNOME Keyring package contains a daemon that keeps passwords and other secrets for users.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-keyring/46/gnome-keyring-46.2.tar.xz>
- Download MD5 sum: 7a8ab16a87f03ca05fc176925fcce649
- Download size: 1.3 MB
- Estimated disk space required: 129 MB
- Estimated build time: 0.2 SBU (Using parallelism=4; add 0.2 SBU for tests)

GNOME Keyring Dependencies

Required

[dbus-1.14.10](#) and [Gcr-3.41.2](#)

Recommended

[Linux-PAM-1.6.1](#), [libxslt-1.1.42](#), and [OpenSSH-9.8p1](#)

Optional

[GnuPG-2.4.5](#), [Valgrind-3.23.0](#), [LCOV](#), and [libcap-ng](#)

Installation of GNOME Keyring

Install GNOME Keyring by running the following commands:

```
sed -i 's:"/desktop:/org:' schema/*.xml &&
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --enable-ssh-agent &&
make
```

A session bus address is necessary to run the tests. To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ... schema/*.xml`: This command fixes a deprecated entry in the schema template.

`--enable-ssh-agent`: This switch enables the SSH agent built in to gnome-keyring instead of the one in GCR. This is done due to bugs in gcr which cause the agent to not work correctly.

Contents

Installed Programs: gnome-keyring (symlink), gnome-keyring-3, and gnome-keyring-daemon

Installed Libraries: gnome-keyring-pkcs11.so (PKCS#11 module) and pam_gnome_keyring.so (PAM module)

Installed Directory: /usr/lib/gnome-keyring and /usr/share/xdg-desktop-portal

Short Descriptions

`gnome-keyring-daemon` is a session daemon that keeps passwords for users

gnome-settings-daemon-46.0

Introduction to GNOME Settings Daemon

The GNOME Settings Daemon is responsible for setting various parameters of a GNOME Session and the applications that run under it.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-settings-daemon/46/gnome-settings-daemon-46.0.tar.xz>
- Download MD5 sum: 408d4fc18cf6971828ff957e052ce4dd
- Download size: 1.4 MB
- Estimated disk space required: 35 MB
- Estimated build time: 0.1 SBU (Using parallelism=4)

GNOME Settings Daemon Dependencies

Required

[alsa-lib-1.2.12](#), [colord-1.4.7](#), [Fontconfig-2.15.0](#), [Gcr-4.3.0](#), [GeoClue-2.7.1](#), [geocode-glib-3.26.4](#), [gnome-desktop-44.1](#), [Little CMS-2.16](#), [libcanberra-0.30](#), [libgweather-4.4.2](#), [libnotify-0.8.3](#), [libwacom-2.12.2](#), [PulseAudio-17.0](#), and [UPower-1.90.4](#)

Recommended

[Cups-2.4.10](#), [NetworkManager-1.48.8](#), [nss-3.103](#), and [Wayland-1.23.0](#)

Note

Recommended dependencies are not strictly required for this package to build and function, but you may not get expected results at runtime if you don't install them.

Optional

[gnome-session-46.0](#), [Mutter-46.4](#), [dbusmock-0.32.1](#), [umockdev-0.18.3](#), and [xvfb](#) (for tests, from [Xorg-Server-21.1.13](#) or [Xwayland-24.1.2](#))

Installation of GNOME Settings Daemon

Install GNOME Settings Daemon by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To check the results, execute: `ninja test`. Note that you must have `python-dbusmock` installed in order for the tests to complete successfully. Some tests may fail depending on the init system in use.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Libraries: libgsd.so

Installed Directories: /etc/xdg/Xwayland-session.d, /usr/include/gnome-settings-daemon-46, /usr/lib/gnome-settings-daemon-46, and /usr/share/gnome-settings-daemon

Tecla-46.0

Introduction to Tecla

The Tecla package contains a keyboard layout viewer.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/tecla/46/tecla-46.0.tar.xz>
- Download MD5 sum: 8e3ecd44ed17dab85aa281df19357395
- Download size: 36 KB
- Estimated disk space required: 2.2 MB
- Estimated build time: less than 0.1 SBU

Tecla Dependencies

Required

[libadwaita-1.5.3](#) and [libxkbcommon-1.7.0](#)

Installation of Tecla

Install Tecla by running the following commands:

```
mkdir build &&
cd build &&
```

```
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: `tecla`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`tecla` is a keyboard layout viewer

gnome-control-center-46.4

Introduction to GNOME Control Center

The GNOME Control Center package contains the GNOME settings manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-control-center/46/gnome-control-center-46.4.tar.xz>
- Download MD5 sum: 661850c7283a0ef1a50efd4184906b10
- Download size: 7.7 MB
- Estimated disk space required: 120 MB (with tests)
- Estimated build time: 0.5 SBU (Using parallelism=4; with tests)

GNOME Control Center Dependencies

Required

[AccountsService-23.13.9](#), [colord-gtk-0.3.1](#), [gnome-online-accounts-3.50.4](#), [gnome-settings-daemon-46.0](#), [gsound-1.0.3](#), [libadwaita-1.5.3](#), [libgtop-2.41.3](#), [libpwquality-1.4.5](#), [MIT Kerberos V5-1.21.3](#), [shared-mime-info-2.4](#), [Tecla-46.0](#), and [UDisks-2.10.1](#)

Recommended

[Cups-2.4.10](#) and [Samba-4.20.4](#) (for the Printers Panel), [gnome-bluetooth-46.1](#), [ibus-1.5.30](#), [ModemManager-1.18.12](#) and [libnm-1.10.6](#) (for the Network Panel)

Optional

`xvfb` (from [Xorg-Server-21.1.13](#) or [Xwayland-24.1.2](#)) and [dbusmock-0.32.1](#) (both for tests)

Optional Runtime Dependencies

[cups-pk-helper-0.2.7](#) (Printers panel), [gnome-color-manager-3.36.0](#) (Color panel), [gnome-shell-46.4](#) (Applications panel), and [sound-theme-freedesktop-0.8](#) (Additional Sound Effects in Sound panel)

Note

Recommended dependencies are not strictly required for this package to build and function, but you may not get expected results at runtime if you don't install them.

Installation of GNOME Control Center

Install GNOME Control Center by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `GTK_A11Y=none ninja test`. Note that you must have the `python-dbusmock` module installed in order for the tests to complete successfully.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D ibus=false`: Use this switch if you have not installed the recommended IBus dependency.

Contents

Installed Programs: `gnome-control-center`

Installed Libraries: None

Installed Directories: /`usr/share/gnome-control-center`, /`usr/share/pixmaps/faces`, and /`usr/share/sounds/gnome`

Short Descriptions

`gnome-control-center` is a graphical user interface used to configure various aspects of GNOME

Mutter-46.4

Introduction to Mutter

Mutter is the window manager for GNOME. It is not invoked directly, but from GNOME Session (on a machine with a hardware accelerated video driver).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/mutter/46/mutter-46.4.tar.xz>
- Download MD5 sum: 022716bf531a4953a74910fef09019ed
- Download size: 6.4 MB
- Estimated disk space required: 83 MB (with tests)
- Estimated build time: 0.5 SBU (Using parallelism=4; add 1.4 SBU for tests)

Mutter Dependencies

Required

`gnome-settings-daemon-46.0`, `graphene-1.10.8`, `libei-1.3.0`, `libxcvt-0.1.2`, `libxkbcommon-1.7.0`, and `pipewire-1.2.3`

Recommended

[desktop-file-utils-0.27](#), [GLib-2.80.4](#) (with GObject Introspection), and [startup-notification-0.12](#)

Recommended (Required to build the Wayland compositor)

[libinput-1.26.1](#), [Wayland-1.23.0](#), [wayland-protocols-1.36](#), and [Xwayland-24.1.2](#)

Optional

[dbusmock-0.32.1](#) (required for tests), [libdisplay-info-0.2.0](#), [Xorg-Server-21.1.13](#) (for X11 sessions), and [sysprof](#)

Installation of Mutter

Install Mutter by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D tests=false     \
            -D profiler=false  \
            ..                 &&
ninja
```

The test suite requires an external program called [xvfb-run](#). If you wish to run the tests, you should download and install it before running `meson`. You can obtain it from [xvfb-run](#), and install it with executable permissions in `/usr/bin`. `xvfb-run` needs `xvfb` at runtime, and `xvfb` can be installed from either [Xorg-Server-21.1.13](#) or [Xwayland-24.1.2](#). You should also replace `-D tests=false` in the `meson` command, with `-D tests=true -D clutter_tests=false`. The test suite requires the mutter schema to be installed on the system, so it is better to run the tests after installing the package.

You can also test basic functions of Mutter following [the section called "Starting Mutter"](#), after installing it.

Now, as the `root` user:

```
ninja install
```

If you wish to run the tests, remove the reference to Zenity which is not a part of BLFS from one test:

```
sed 's/zenity --[a-z]*/gtk4-demo/' -i .../src/tests/x11-test.sh
```

Now run the test suite:

```
ninja test
```

The tests require an active X or wayland session. Some tests are flaky (especially under a high system load) so if a test fails you can try to re-run it alone with the `meson test <test name>` command. A few tests may fail depending on some system configuration. Don't make any mouse or keyboard input while the test suite is running or some tests may fail.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D tests=false`: Prevents building the tests, and removes a hard requirement on `xvfb-run`.

`-D profiler=false`: Allows building this package without Sysprof. Remove this option if you've installed Sysprof and want to analyze the rendering performance of Mutter.

`-D tests=true -D clutter_tests=false`: This builds all of the tests except for the tests for the shipped Clutter library. The Clutter tests are known to fail with `--buildtype=release`.

Starting Mutter

Mutter is normally used as a component of gnome-shell, but it can be used as a standalone Wayland compositor too. To run Mutter as a Wayland compositor, in a virtual console, issue:

```
mutter --wayland -- vte-2.91
```

Replace `vte-2.91` with the command line for the first application you want in the Wayland session. Note that once this application exits, the Wayland session will be terminated.

Mutter can also function as a nested compositor in another Wayland session. In a terminal emulator, issue:

```
MUTTER_DEBUG_DUMMY_MODE_SPECS=1920x1080 mutter --wayland --nested -- vte-2.91
```

Replace `1920x1080` with the size you want for the nested Wayland session.

Contents

Installed Programs: mutter

Installed Libraries: libmutter-14.so and libmutter-test-14.so (optional)

Installed Directories: /usr/{lib,include,libexec/installed-tests,share/{,installed-tests}}/mutter-14

Short Descriptions

<code>mutter</code>	is a Clutter based compositing GTK+ Window Manager
<code>libmutter-14.so</code>	contains the Mutter API functions
<code>libmutter-test-14.so</code>	contains the Mutter test suite API functions; this library is only installed if the test suite of this package is enabled and it is needed by gnome-shell test suite

gnome-shell-46.4

Introduction to GNOME Shell

The GNOME Shell is the core user interface of the GNOME Desktop environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-shell/46/gnome-shell-46.4.tar.xz>
- Download MD5 sum: 44b69588f138be1e70fbe307eff3346f
- Download size: 2.0 MB
- Estimated disk space required: 62 MB (with tests)
- Estimated build time: 0.3 SBU (Using parallelism=4; with tests)

GNOME Shell Dependencies

Required

[evolution-data-server-3.52.4](#), [Gcr-4.3.0](#), [Gjs-1.80.2](#), [gnome-desktop-44.1](#), [ibus-1.5.30](#), [Mutter-46.4](#), [Polkit-125](#), [sassc-3.6.2](#), and [startup-notification-0.12](#)

Recommended

[desktop-file-utils-0.27](#), [gnome-autoar-0.4.4](#), [gnome-bluetooth-46.1](#), [gst-plugins-base-1.24.7](#), [NetworkManager-1.48.8](#), and [power-profiles-daemon-0.21](#)

Optional

[GTK-Doc-1.34.0](#) and [bash-completion](#)

Required Runtime Dependencies

[adwaita-icon-theme-46.2](#), [DConf-0.40.0](#), [GDM-46.2](#), [gnome-backgrounds-46.0](#), [gnome-control-center-46.4](#), [gnome-menus-3.36.0](#), [libgweather-4.4.2](#), and [Systemd-256.4](#)

Installation of GNOME Shell

Install GNOME Shell by running the following commands:

```
mkdir build &&
cd      build &&
```

```
meson setup --prefix=/usr \
            --buildtype=release \
            -D tests=false \
            .. \
            &&
ninja
```

To test the results, you need to remove the `-D tests=false` switch, and to have [Mutter-46.4](#) compiled and installed with tests also. To run the tests, issue: `ninja test`. You must be running an X session to run the tests. Some will fail if [GDM-46.2](#) is not installed. One test named `css styling support` is known to fail with some multi-monitor configurations.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D tests=false`: Remove this switch if you want to build the tests. However, `meson` will fail if the [Mutter-46.4](#) tests are not installed.

`-D extensions_tool=false`: This switch allows building this package without [gnome-autoar-0.4.4](#) installed.

Contents

Installed Programs: gnome-extensions, gnome-extensions-app, gnome-shell, gnome-shell-extension-prefs, gnome-shell-extension-tool, and gnome-shell-test-tool

Installed Libraries: None

Installed Directories: /usr/lib/gnome-shell, /usr/share/gnome-shell, /usr/share/gtk-doc/html/{shell,st} (optional), and /usr/share/xdg-desktop-portal/

Short Descriptions

`gnome-shell` provides the core user interface functions for the GNOME Desktop

gnome-shell-extensions-46.2

Introduction to GNOME Shell Extensions

The GNOME Shell Extensions package contains a collection of extensions providing additional and optional functionality to the GNOME Shell.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-shell-extensions/46/gnome-shell-extensions-46.2.tar.xz>
- Download MD5 sum: 915c2b4fd0c64ec6ef00029ef3edf3bb
- Download size: 220 KB
- Estimated disk space required: 3.7 MB
- Estimated build time: less than 0.1 SBU

GNOME Shell Extensions Dependencies

Required

[libgtop-2.41.3](#)

Optional

[sassc-3.6.2](#)

Installation of GNOME Shell Extensions

Install GNOME Shell Extensions by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr ..
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/gnome-shell/extensions

gnome-session-46.0

Introduction to GNOME Session

The GNOME Session package contains the GNOME session manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-session/46/gnome-session-46.0.tar.xz>
- Download MD5 sum: f68f71cb53d650916f9ef9cb6076f643
- Download size: 476 KB
- Estimated disk space required: 14 MB
- Estimated build time: 0.2 SBU

GNOME Session Dependencies

Required

[gnome-desktop-44.1](#), [JSON-GLib-1.8.0](#), [Mesa-24.1.5](#), [Systemd-256.4](#) (runtime), and [UPower-1.90.4](#)

Optional

[xmlto-0.0.29](#), and [libxslt-1.1.42](#) with [docbook-xml-4.5](#) and [docbook-xsl-nons-1.79.2](#) (to build the documentation)

Installation of GNOME Session

When running GNOME under [Wayland-1.23.0](#), environment settings are not imported for the user using the system profile. The Wayland developers are currently undecided on a standard method to provide system environment settings for user sessions. To work around this limitation, execute the following command to make `gnome-session` use a login shell:

```
sed 's@/bin/sh@/bin/sh -l@' -i gnome-session/gnome-session.in
```

Install GNOME Session by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Move the documentation to a versioned directory:

```
mv -v /usr/share/doc/gnome-session{,-46.0}
```

This package creates two Xorg based .desktop files in the /usr/share/xsessions/ directory, and two Wayland based .desktop files in the /usr/share/wayland-sessions/ directory. Only one is needed in each directory on a BLFS system, so prevent the extra files showing up as options in a display manager. As the `root` user:

```
rm -v /usr/share/xsessions/gnome.desktop &&  
rm -v /usr/share/wayland-sessions/gnome.desktop
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Starting GNOME

The easiest way to start GNOME is to use a display manager. [GDM-46.2](#) is the recommended display manager.

It is also possible to start GNOME from the command line. However, you must still have [GDM-46.2](#) installed for some portions of the desktop to work. One reason to start from the command line is if you want Wayland support on a system that has the proprietary NVIDIA driver installed, since GDM will not show the Wayland session type on such a system.

To start GNOME using [xinit-1.4.2](#), run the following commands:

```
cat > ~/.xinitrc << "EOF"  
dbus-run-session gnome-session  
EOF  
  
startx
```

Alternatively, to start GNOME with Wayland support, run the following command:

```
XDG_SESSION_TYPE=wayland dbus-run-session gnome-session
```

Contents

Installed Programs: gnome-session, gnome-session-inhibit, and gnome-session-quit

Installed Libraries: None

Installed Directories: /usr/share/doc/gnome-session-46.0 and /usr/share/gnome-session

Short Descriptions

<code>gnome-session</code>	is used to start up the GNOME Desktop environment
<code>gnome-session-inhibit</code>	is used to inhibit certain GNOME Session functionality while executing the given command
<code>gnome-session-quit</code>	is used to end the GNOME Session

gnome-tweaks-46.1

Introduction to GNOME Tweaks

GNOME Tweaks is a simple program used to tweak advanced GNOME settings.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-tweaks/46/gnome-tweaks-46.1.tar.xz>
- Download MD5 sum: 1c770b4db46b037338e0501a6c5cc813
- Download size: 676 KB
- Estimated disk space required: 4.4 MB

- Estimated build time: less than 0.1 SBU

GNOME Tweaks Dependencies

Required

[GTK-4.14.5](#), [gsettings-desktop-schemas-46.1](#), [libadwaita-1.5.3](#), [libgudev-238](#), [PyGObject-3.48.2](#), and [sound-theme-freedesktop-0.8](#)

Installation of GNOME Tweaks

Install GNOME Tweaks by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: gnome-tweaks

Installed Libraries: None

Installed Directories: /usr/lib/python3.12/site-packages/gtweak and /usr/share/gnome-tweaks

Short Descriptions

`gnome-tweaks` is used to tweak advanced GNOME settings

gnome-user-docs-46.4

Introduction to GNOME User Docs

The GNOME User Docs package contains documentation for GNOME.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-user-docs/46/gnome-user-docs-46.4.tar.xz>
- Download MD5 sum: 1fbe483547bea3110458b9323bc9dc95
- Download size: 13 MB
- Estimated disk space required: 166 MB
- Estimated build time: 0.5 SBU (using parallelism=4)

GNOME User Docs Dependencies

Required

[itstool-2.0.7](#) and [libxml2-2.13.3](#)

Installation of GNOME User Docs

Install GNOME User Docs by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/help/*/gnome-help and /usr/share/help/*/system-admin-guide

Yelp-42.2

Introduction to Yelp

The Yelp package contains a help browser used for viewing help files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/yelp/42/yelp-42.2.tar.xz>
- Download MD5 sum: 3792122c4ab90725716cd88e9274f0f6
- Download size: 1.4 MB
- Estimated disk space required: 21 MB
- Estimated build time: 0.1 SBU

Yelp Dependencies

Required

[gsettings-desktop-schemas-46.1](#), [WebKitGTK-2.44.3](#) and [yelp-xsl-42.1](#)

Recommended

[desktop-file-utils-0.27](#)

Optional

[GTK-Doc-1.34.0](#)

Note

The Yelp package is not required for a functional GNOME Desktop. Note, however, that without Yelp you will not be able to view the built-in Help provided by core GNOME and many of the support applications.

Installation of Yelp

Install Yelp by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Still as the `root` user, ensure that `/usr/share/applications/mimeinfo.cache` will be updated and therefore the `help` in gnome applications will work.

```
update-desktop-database
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Program: `gnome-help` (symlink) and `yelp`

Installed Library: `libyelp.so`

Installed Directories: `/usr/include/libyelp`, `/usr/share/gtk-doc/html/libyelp` (optional), and `/usr/{lib,share}/yelp`

Short Descriptions

`yelp` is the GNOME Help Browser

`libyelp.so` contains the Yelp API functions

Chapter 34. GNOME Applications

These packages are desktop applications and assorted utilities from the GNOME project. Feel free to install them on an as needed or as desired basis.

Baobab-46.0

Introduction to Baobab

The Baobab package contains a graphical directory tree analyzer.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/baobab/46/baobab-46.0.tar.xz>
- Download MD5 sum: 9d6749f5c88db72c2fcc421b65aa6630
- Download size: 592 KB
- Estimated disk space required: 12 MB
- Estimated build time: less than 0.1 SBU (Using parallelism=4)

Baobab Dependencies

Required

[adwaita-icon-theme-46.2](#), [GTK-4.14.5](#), [itstool-2.0.7](#), [libadwaita-1.5.3](#), and [Vala-0.56.17](#)

Installation of Baobab

Install Baobab by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: baobab

Installed Libraries: None

Installed Directories: /usr/share/help/*/baobab

Short Descriptions

`baobab` is a graphical tool used to analyze disk usage

Brasero-3.12.3

Introduction to Brasero

Brasero is an application used to burn CDs and DVDs on the GNOME Desktop. It is designed to be as simple as possible and has some unique features that enable users to create their discs easily and quickly.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/brasero/3.12/brasero-3.12.3.tar.xz>
- Download MD5 sum: ae48248dd36f89282d573eb7a0a1391f
- Download size: 3.0 MB
- Estimated disk space required: 105 MB
- Estimated build time: 1.0 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/brasero-3.12.3-upstream_fixes-1.patch

Brasero Dependencies

Required

[gst-plugins-base-1.24.7](#), [itstool-2.0.7](#), [libcanberra-0.30](#), and [libnotify-0.8.3](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [libburn-1.5.6](#), [libisoburn-1.5.6](#), and [libisofs-1.5.6](#), [Nautilus-46.2](#), [Tracker-3.7.3](#), and [totem-pl-parser-3.26.6](#)

Optional

[GTK-Doc-1.34.0](#)

Recommended (Runtime)

[dvd+rw-tools-7.1](#) and [Gvfs-1.54.2](#)

Optional (Runtime)

[Cdrdao-1.2.4](#), [libdvdcss-1.4.3](#), [Cdrtools-3.02a09](#), and [VCDImager](#)

Installation of Brasero

First, fix a build issue that occurs with GCC-14:

```
patch -Np1 -i ../brasero-3.12.3-upstream_fixes-1.patch
```

Install Brasero by running the following commands:

```
./configure --prefix=/usr \
--enable-compile-warnings=no \
--enable-cxx-warnings=no \
--disable-nautilus \
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-*-warnings=no`: These switches fix build problems under GCC-6.x compilers.

`--disable-nautilus`: This switch disables building the Nautilus extension since it is incompatible with the gtk4 version of Nautilus.

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: brasero

Installed Libraries: libbrasero-{burn,media,utils}3.so

Installed Directories: /usr/include/brasero3, /usr/lib/brasero3, /usr/share/brasero, /usr/share/gtk-doc/html/libbrasero-{burn,media}, and /usr/share/help/*/brasero

Short Descriptions

<code>brasero</code>	is a simple and easy to use CD/DVD burning application for the GNOME Desktop
<code>libbrasero-burn3.so</code>	contains the Burning API functions
<code>libbrasero-media3.so</code>	contains the Media API functions
<code>libbrasero-utils3.so</code>	contains the Brasero API functions

EOG-45.4

Introduction to EOG

EOG is an application used for viewing and cataloging image files on the GNOME Desktop. It also has basic editing capabilities.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/eog/45/eog-45.4.tar.xz>
- Download MD5 sum: 7de057cbfe343d32b2cc5b7792431edd
- Download size: 4.4 MB
- Estimated disk space required: 32 MB
- Estimated build time: 0.3 SBU

EOG Dependencies

Required

[adwaita-icon-theme-46.2](#), [Exempi-2.6.5](#), [gnome-desktop-44.1](#), [libhandy-1.8.3](#), [libjpeg-turbo-3.0.1](#), [libpeas-1.36.0](#), and [shared-mime-info-2.4](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection), [Little CMS-2.16](#), [libexif-0.6.24](#), [librsvg-2.58.3](#), and [webp-pixbuf-loader-0.2.7](#)

Optional

[GTK-Doc-1.34.0](#) and [libportal-0.7.1](#)

Installation of EOG

Install EOG by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D libportal=false \
            ..
            &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Still as the `root` user, rebuild the cache database of MIME types handled by desktop files so [Nautilus-46.2](#) will be able to open image files with EOG.

```
update-desktop-database
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D libportal=false`: This parameter disables building EOG with support for flatpak portals. Remove this parameter if you have [libportal-0.7.1](#) installed and wish to have support for flatpak portals.

`-D gtk_doc=true`: Use this option if [GTK-Doc-1.34.0](#) is installed and you wish to rebuild and install the API documentation.

Contents

Installed Program: eog

Installed Library: libeog.so

Installed Directories: /usr/include/eog-3.0, /usr/lib/eog, /usr/share/eog, /usr/share/gtk-doc/html/eog (optional), and /usr/share/help/*/eog

Short Descriptions

`eog` is a fast and functional image viewer as well as an image cataloging program. It has basic editing capabilities

Evince-46.3.1

Introduction to Evince

Evince is a document viewer for multiple document formats. It supports PDF, Postscript, DjVu, TIFF and DVI. It is useful for viewing documents of various types using one simple application instead of the multiple document viewers that once existed on the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/evince/46/evince-46.3.1.tar.xz>
- Download MD5 sum: e017e3544751c2442020f4d10a39e24d
- Download size: 2.8 MB
- Estimated disk space required: 49 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

Evince Dependencies

Required

[adwaita-icon-theme-46.2](#), [gsettings-desktop-schemas-46.1](#), [GTK+-3.24.43](#), [itstool-2.0.7](#), [libhandy-1.8.3](#), [libxml2-2.13.3](#), and [OpenJPEG-2.5.2](#)

Recommended

[gnome-keyring-46.2](#), [GLib-2.80.4](#) (with GObject Introspection), [libarchive-3.7.4](#), [libsecret-0.21.4](#), and [Poppler-24.08.0](#)

Optional

[Cups-2.4.10](#) (to enable printing if support is built into GTK+ 3), [gnome-desktop-44.1](#), [gspell-1.12.2](#), [gst-plugins-base-1.24.7](#), [Gi-DocGen-2024.1](#), [libgxpath-0.3.2](#), [libtiff-4.6.0](#), [texlive-20240312](#) (or [install-tl-unx](#)), [DjVuLibre](#), [libspectre](#), and [Synctex](#)

Installation of Evince

Install Evince by running the following commands:

```
mkdir build &&
cd build &&

CPPFLAGS+=" -I/opt/texlive/2024/include" \
meson setup --prefix=/usr \
--buildtype=release \
-D gtk_doc=false \
--wrap-mode=nodownload \
..
&&
ninja
```

If you have [Gi-DocGen-2024.1](#) installed and wish to build the API documentation for this package, issue:

```
sed "/docs_dir/s@\$@ / 'evince-46.3.1'@ -i ./help/meson.build &&
meson configure -D gtk_doc=true &&
ninja
```

This package does not have a working test suite.

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a “DESTDIR” method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`CPPFLAGS+=" -I/opt/texlive/2024/include"`: Make sure that meson can find libkpathsea headers from TeXLive if it is installed. This has no effect on systems without TeXLive installed.

`--wrap-mode=nodownload`: This switch disables fetching any dependency: the build system tries to fetch gi-docgen even if `-D gtk_doc=false` is passed.

`-D gtk_doc=false`: Allow building this package without [Gi-DocGen-2024.1](#) installed. If you have [Gi-DocGen-2024.1](#) installed and you wish to rebuild and install the API documentation, a `meson configure` command will reset this option.

`-D gspell=false`: This switch turns off support for the spell checking plugin.

`-D introspection=false`: Use this option if you don't have built [GLib-2.80.4](#) with GObject Introspection and don't wish to have introspection support built into Evince.

`-D nautilus=false`: This switch disables building the Nautilus Plugin. Use this switch if Nautilus is not installed.

`-D keyring=false`: This switch disables the use of libsecret. Use this switch if libsecret is not installed.

`-D ps=enabled`: Use this switch if libspectre is installed and you want to view PostScript files with Evince.

Contents

Installed Programs: evince, evince-previewer, and evince-thumbnailer

Installed Libraries: libevdocument3.so and libevview3.so

Installed Directories: /usr/{include,lib,share}/evince, /usr/share/gtk-doc/html/{evince,libevdocument-3.0,libevview-3.0} (optional), and /usr/share/help/*/evince

Short Descriptions

`evince` is a multiple format document viewer

`evince-previewer` is an application that implements the printing previewer

`evince-thumbnailer` is a simple program used to create thumbnail images of supported documents

Evolution-3.52.4

Introduction to Evolution

The Evolution package contains an integrated mail, calendar and address book suite designed for the GNOME environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/evolution/3.52/evolution-3.52.4.tar.xz>
- Download MD5 sum: b4acefb4587e7a6bd93a1046419b4705
- Download size: 13 MB
- Estimated disk space required: 380 MB
- Estimated build time: 1.0 SBU (Using parallelism=4)

Evolution Dependencies

Required

[adwaita-icon-theme-46.2](#), [evolution-data-server-3.52.4](#), [Gcr-4.3.0](#), [gnome-autoar-0.4.4](#), [shared-mime-info-2.4](#), and [WebKitGTK-2.44.3](#)

Recommended

[Bogofilter-1.2.5](#), [enchant-2.8.2](#), [gnome-desktop-44.1](#), [gspell-1.12.2](#), [Highlight-4.13](#), [itstool-2.0.7](#), [libcanberra-0.30](#), [libgweather-4.4.2](#), [libnotify-0.8.3](#), [OpenLDAP-2.6.8](#), and [Seahorse-43.0](#)

Optional

[GeoClue-2.7.1](#), [geocode-glib-3.26.4](#), and [GTK-Doc-1.34.0](#), [clutter-gtk](#) (Contact Maps plugin), [cmark](#), [Glade](#), [libchamplain](#) (Contact Maps plugin), [libpst](#), [libunify](#), [libytnef](#)

Installation of Evolution

Install Evolution by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D SYSCONF_INSTALL_DIR=/etc \
-D ENABLE_INSTALLED_TESTS=ON \
-D ENABLE_PST_IMPORT=OFF \
-D ENABLE_YTNEF=OFF \
-D ENABLE_CONTACT_MAPS=OFF \
-D ENABLE_MARKDOWN=OFF \
-D ENABLE_WEATHER=ON \
-G Ninja .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D ENABLE_YTNEF=OFF`: This switch is used to disable the yTNEF library usage because libytnef is not part of BLFS.

`-D ENABLE_PST_IMPORT=OFF`: This switch is used to disable the pst-import plugin because libpst is not part of BLFS.

`-D ENABLE_CONTACT_MAPS=OFF`: This switch disables building the Contact Maps plugin. Remove this switch if you have installed the necessary dependencies and wish to build the Contact Maps plugin.

`-D ENABLE_MARKDOWN=OFF`: This switch allows building without [cmark](#). Remove this switch if you need markdown support and you have installed the necessary dependency.

`-D ENABLE_WEATHER=ON`: This switch allows building against [libgweather-4.4.2](#).

`-D WITH_HELP=OFF`: This switch disables building the manual of this package. Use this switch if you have not installed [itstool-2.0.7](#).

`-D WITH_OPENLDAP=OFF`: Use this switch if you have not installed [OpenLDAP-2.6.8](#).

Contents

Installed Programs: evolution

Installed Libraries: None

Installed Directories: /usr/include/evolution, /usr/lib/evolution, /usr/libexec/evolution, /usr/share/evolution, /usr/share/installed-tests/evolution, /usr/share/help/*/evolution, and optionally /usr/share/gtk-doc/html/{evolution-mail-composer,evolution-mail-engine}, /usr/share/gtk-doc/html/{evolution-mail-formatter,evolution-shell}, and /usr/share/gtk-doc/html/evolution-util

Short Descriptions

`evolution` is an email, calendar and address book suite for the GNOME Desktop

File-Roller-44.3

Introduction to File Roller

File Roller is an archive manager for GNOME with support for tar, bzip2, gzip, zip, jar, compress, lzop, zstd, dmg, and many other archive formats.

This package is known to build and work properly using an LFS 12.2 platform.

Note

File Roller is only a graphical interface to archiving utilities such as tar and zip.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/file-roller/44/file-roller-44.3.tar.xz>
- Download MD5 sum: 8f309a2af9788b2291512bdf0cd76ffd
- Download size: 1020 KB
- Estimated disk space required: 21 MB
- Estimated build time: 0.1 SBU (Using parallelism=4)

File Roller Dependencies

Required

[GTK-4.14.5](#) and [itstool-2.0.7](#)

Recommended

[cpio-2.15](#), [desktop-file-utils-0.27](#), [JSON-GLib-1.8.0](#), [libarchive-3.7.4](#), [libadwaita-1.5.3](#), [libportal-0.7.1](#), and [Nautilus-46.2](#)

Optional (for the API documentation)

[Gi-DocGen-2024.1](#)

Optional (Runtime)

[UnRAR-7.0.9](#), [UnZip-6.0](#), and [Zip-3.0](#)

Installation of File Roller

Install File Roller by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D packagekit=false \
            ..
            &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install &&
chmod -v 0755 /usr/libexec/file-roller/isoinfo.sh
```

Note

If you installed the package to your system using a "DESTDIR" method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+-3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

-D packagekit=false: This switch disables the use of PackageKit which isn't suitable for BLFS.

-D api_docs=enabled: Use this switch if you have [Gi-DocGen-2024.1](#) installed and wish to generate the API documentation.

Contents

Installed Program: file-roller

Installed Libraries: libnautilus-fileroller.so

Installed Directories: /usr/{libexec,share{./help/*}}/file-roller

Short Descriptions

file-roller is an archive manager for GNOME

gnome-calculator-46.1

Introduction to GNOME Calculator

GNOME Calculator is a powerful graphical calculator with financial, logical and scientific modes. It uses a multiple precision package to do its arithmetic to give a high degree of accuracy.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-calculator/46/gnome-calculator-46.1.tar.xz>
- Download MD5 sum: e3170847e49203e645abc3f3c91224a6
- Download size: 1.1 MB
- Estimated disk space required: 43 MB (with tests)
- Estimated build time: 0.4 SBU (with tests)

GNOME Calculator Dependencies

Required

[gtksourceview5-5.12.1](#), [itstool-2.0.7](#), [libadwaita-1.5.3](#), [libgee-0.20.6](#), and [libsoup-3.4.4](#)

Recommended

[Vala-0.56.17](#)

Installation of GNOME Calculator

Install GNOME Calculator by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: gcalccmd and gnome-calculator

Installed Library: libgcalc-2.so and libgci-1.so

Installed Directories: /usr/include/gcalc-2, /usr/include/gci-2, /usr/share/devhelp/books/{GCalc-2,GCi-1}, and /usr/share/help/*/gnome-calculator

Short Descriptions

gnome-calculator is the official calculator of the GNOME Desktop
gcalccmd is a command line version of gnome-calculator

gnome-color-manager-3.36.0

Introduction to GNOME Color Manager

GNOME Color Manager is a session framework for the GNOME desktop environment that makes it easy to manage, install and generate color profiles.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-color-manager/3.36/gnome-color-manager-3.36.0.tar.xz>
- Download MD5 sum: c5360705a68e88455b1801200e9aaa2e
- Download size: 2.1 MB
- Estimated disk space required: 12 MB
- Estimated build time: 0.1 SBU

GNOME Color Manager Dependencies

Required

[colord-1.4.7](#), [GTK+-3.24.43](#), [itstool-2.0.7](#), [Little CMS-2.16](#), [libcanberra-0.30](#), and [libexif-0.6.24](#)

Recommended

[desktop-file-utils-0.27](#)

Optional

[appstream-glib-0.8.3](#) and [DocBook-utils-0.6.14](#) (currently causes build to fail)

Installation of GNOME Color Manager

If [DocBook-utils-0.6.14](#) is installed, disable installation of the man pages to avoid a build failure:

```
sed /subdir\(\`man/d -i meson.build
```

Install GNOME Color Manager by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`. The tests must be run from an X session.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: gcm-import, gcm-inspect, gcm-picker, and gcm-viewer

Installed Libraries: None

Installed Directories: /usr/share/gnome-color-manager and /usr/share/help/*/gnome-color-manager

Short Descriptions

gcm-import	allows you to import ICC profiles supplied by vendors
gcm-inspect	allows you to inspect your session color-management settings
gcm-picker	allows you to pick spot colors for use by an attached colorimeter
gcm-viewer	allows you to view properties of ICC profiles

gnome-connections-46.0

Introduction to gnome-connections

gnome-connections is a VNC and RDP client for the GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-connections/46/gnome-connections-46.0.tar.xz>
- Download MD5 sum: dcd819bfb15af4652842fb6b91fb9c7f
- Download size: 3.9 MB
- Estimated disk space required: 16 MB
- Estimated build time: 0.1 SBU

gnome-connections Dependencies

Required

[AppStream-1.0.3](#), [FreeRDP-3.7.0](#), [gtk-vnc-1.3.1](#), [itstool-2.0.7](#), [libhandy-1.8.3](#), and [Vala-0.56.17](#)

Installation of gnome-connections

Install gnome-connections by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: gnome-connections

Installed Libraries: None

Installed Directories: /usr/include/gnome-connections, /usr/lib/gnome-connections, /usr/share/gnome-connections, and /usr/share/help/*/gnome-connections

Short Descriptions

gnome-disk-utility-46.0

Introduction to GNOME Disk Utility

The GNOME Disk Utility package provides applications used for dealing with storage devices.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-disk-utility/46/gnome-disk-utility-46.0.tar.xz>
- Download MD5 sum: e1ba191bbc9c38e99f6710520fbb5a20
- Download size: 1.7 MB
- Estimated disk space required: 36 MB
- Estimated build time: 0.2 SBU

GNOME Disk Utility Dependencies

Required

[gnome-settings-daemon-46.0](#), [itstool-2.0.7](#), [libdvdread-6.1.3](#), [libhandy-1.8.3](#), [libpwquality-1.4.5](#), [libsecret-0.21.4](#), and [UDisks-2.10.1](#)

Optional

[appstream-glib-0.8.3](#)

Installation of GNOME Disk Utility

Install GNOME Disk Utility by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: gnome-disk-image-mount and gnome-disks

Installed Libraries: None

Installed Directories: None

Short Descriptions

`gnome-disk-image-mount`

is used to set up disk images

`gnome-disks`

is used to inspect, format, partition and configure disks and block devices

gnome-logs-45.0

Introduction to GNOME Logs

The GNOME Logs package contains a log viewer for the systemd journal.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-logs/45/gnome-logs-45.0.tar.xz>
- Download MD5 sum: 5b9396cf24528ae72eb4c48934f60df2
- Download size: 624 KB
- Estimated disk space required: 8.8 MB
- Estimated build time: 0.1 SBU

GNOME Logs Dependencies

Required

[GTK-4.14.5](#), [gsettings-desktop-schemas-46.1](#), [itstool-2.0.7](#), and [libadwaita-1.5.3](#)

Optional

[appstream-glib-0.8.3](#), [desktop-file-utils-0.27](#) (for the test suite), and [docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), [libxslt-1.1.42](#) (to build manual pages), and [dogtail](#)

Installation of GNOME Logs

Install GNOME Logs by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: `gnome-logs`

Installed Libraries: None

Installed Directories: /usr/share/gnome-logs and /usr/share/help/*/gnome-logs

Short Descriptions

`gnome-logs` is a GNOME log viewer for the systemd journal

gnome-maps-46.11

Introduction to GNOME Maps

GNOME Maps is a map application for GNOME.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-maps/46/gnome-maps-46.11.tar.xz>
- Download MD5 sum: a85532c578c0c2f1d009f4e32fe6d817
- Download size: 3.1 MB

- Estimated disk space required: 18 MB
- Estimated build time: 0.1 SBU

GNOME Maps Dependencies

Required

[desktop-file-utils-0.27](#), [GeoClue-2.7.1](#), [geocode-glib-3.26.4](#), [Gjs-1.80.2](#), [libadwaita-1.5.3](#), [libportal-0.7.1](#), [libshumate-1.2.3](#), [libgweather-4.4.2](#), and [rest-0.9.1](#)

Installation of GNOME Maps

Install GNOME Maps by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: `gnome-maps` (symlink)

Installed Libraries: `libgnome-maps.so`

Installed Directories: `/usr/lib/gnome-maps` and `/usr/share/gnome-maps`

Short Descriptions

`gnome-maps` is a map application for GNOME

gnome-nettool-42.0

Introduction to GNOME Nettool

The GNOME Nettool package is a network information tool which provides GUI interface for some of the most common command line network tools.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-nettool/42/gnome-nettool-42.0.tar.xz>
- Download MD5 sum: ba99489e9e3a1af03e9f2719acac7beb
- Download size: 413 KB
- Estimated disk space required: 12 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/gnome-nettool-42.0-ping_and_netstat_fixes-1.patch

GNOME Nettool Dependencies

Required

[GTK+-3.24.43](#), [itstool-2.0.7](#), and [libgtop-2.41.3](#)

Runtime Dependencies

Installation of GNOME Nettool

First, adapt GNOME Nettool to changes in the ping, ping6, and netstat utilities:

```
patch -Np1 -i ../gnome-nettool-42.0-ping_and_netstat_fixes-1.patch
```

Then add a fix for newer versions of meson:

```
sed -i '/merge_file/s/(.*// /' data/meson.build
```

Install GNOME Nettool by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Program: gnome-nettool

Installed Libraries: None

Installed Directories: /usr/share/gnome-nettool and /usr/share/help/*/gnome-nettool

Short Descriptions

`gnome-nettool` is a network information tool

gnome-power-manager-43.0

Introduction to GNOME Power Manager

The GNOME Power Manager package contains a tool used to report on power management on the system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-power-manager/43/gnome-power-manager-43.0.tar.xz>
- Download MD5 sum: c7e33249b59c6082312909f65739912e
- Download size: 376 KB
- Estimated disk space required: 7.0 MB
- Estimated build time: less than 0.1 SBU (with tests)

GNOME Power Manager Dependencies

Required

[GTK+-3.24.43](#) and [UPower-1.90.4](#)

Optional

[appstream-glib-0.8.3](#) and [DocBook-utils-0.6.14](#)

Installation of GNOME Power Manager

Install GNOME Power Manager by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: `gnome-power-statistics`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`gnome-power-statistics` is used to visualize the power consumption of laptop hardware

gnome-screenshot-41.0

Introduction to GNOME Screenshot

The GNOME Screenshot is a utility used for taking screenshots of the entire screen, a window or a user-defined area of the screen, with optional beautifying border effects.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-screenshot/41/gnome-screenshot-41.0.tar.xz>
- Download MD5 sum: 3d0199188d8ba07399e468a61e0174e8
- Download size: 352 KB
- Estimated disk space required: 7.4 MB
- Estimated build time: less than 0.1 SBU

GNOME Screenshot Dependencies

Required

[GTK+-3.24.43](#), [libcanberra-0.30](#) (Compiled with GTK+3 support), and [libhandy-1.8.3](#)

Installation of GNOME Screenshot

First, fix building with newer versions of meson:

```
sed -i '/merge_file/{n;d}' data/meson.build
```

Install GNOME Screenshot by running the following commands:

```
mkdir build &&
cd build &&
```

```
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Note

In order to best run GNOME Screenshot from the command line, the `-i` option needs to be specified.

Contents

Installed Program: `gnome-screenshot`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`gnome-screenshot` is used to capture the screen, a window, or a user-defined area and save the snapshot image to a file

gnome-system-monitor-46.0

Introduction to GNOME System Monitor

The GNOME System Monitor package contains GNOME's replacement for `gtop`.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-system-monitor/46/gnome-system-monitor-46.0.tar.xz>
- Download MD5 sum: d2c121f58825bf5a441f16f13b0d3201
- Download size: 1020 KB
- Estimated disk space required: 27 MB
- Estimated build time: 0.2 SBU (with parallelism=4)

GNOME System Monitor Dependencies

Required

[adwaita-icon-theme-46.2](#), [Gtkmm-4.14.0](#), [itstool-2.0.7](#), [libgtop-2.41.3](#), [libadwaita-1.5.3](#), and [librsvg-2.58.3](#)

Optional

[appstream-glib-0.8.3](#), [desktop-file-utils-0.27](#), and [uncrustify](#)

Installation of GNOME System Monitor

Install GNOME System Monitor by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Program: gnome-system-monitor

Installed Libraries: None

Installed Directories: /usr/{libexec,share,share/help/*}/gnome-system-monitor

Short Descriptions

`gnome-system-monitor` is used to display the process tree and hardware meters

gnome-terminal-3.52.2

Introduction to GNOME Terminal

The GNOME Terminal package contains the terminal emulator for GNOME Desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.gnome.org/GNOME/gnome-terminal/-/archive/3.52.2/gnome-terminal-3.52.2.tar.gz>
- Download MD5 sum: ba1805cdc363c92004d6df7962f20dfb
- Download size: 2.8 MB
- Estimated disk space required: 32 MB
- Estimated build time: 0.3 SBU

GNOME Terminal Dependencies

Required

`DConf-0.40.0`, `gnome-shell-46.4`, `gsettings-desktop-schemas-46.1`, `itstool-2.0.7`, `libhandy-1.8.3`, `pcre2-10.44`, and `VTE-0.76.4`

Recommended

`Nautilus-46.2`

Optional

`appstream-glib-0.8.3` and `desktop-file-utils-0.27`

Installation of GNOME Terminal

First, fix some deprecated schema entries:

```
sed -i -r 's:"(/system)":"/org/gnome\1:g' src/external.gschema.xml
```

Install GNOME Terminal by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Note

To run GNOME Terminal, the environment variable `LANG` must be set to a UTF-8 locale *prior* to starting the graphical environment.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D search_provider=false`: This switch disables the "search gnome-shell" provider. Use this option if you do not have gnome-shell installed.

`-D nautilus_extension=false`: This switch disables a dependency on the nautilus file manager. Use this option if you do not have Nautilus installed.

Contents

Installed Program: `gnome-terminal`

Installed Libraries: `/usr/lib/nautilus/extensions-4/libterminal-nautilus.so` (Nautilus Extension)

Installed Directories: `/usr/{lib,share/help/*}/gnome-terminal` and `/usr/share/xdg-terminals`

Short Descriptions

`gnome-terminal` is the GNOME Terminal Emulator

gnome-weather-46.0

Introduction to GNOME Weather

GNOME Weather is a small application that allows you to monitor the current weather conditions for your city, or anywhere in the world, and to access updated forecasts provided by various internet services.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnome-weather/46/gnome-weather-46.0.tar.xz>
- Download MD5 sum: 08c626cea07068b2a02d5faba90fdc14
- Download size: 236 KB
- Estimated disk space required: 4.5 MB
- Estimated build time: less than 0.1 SBU

GNOME Weather Dependencies

Required

[GeoClue-2.7.1](#), [Gjs-1.80.2](#), [libadwaita-1.5.3](#), and [libgweather-4.4.2](#)

Optional

[appstream-glib-0.8.3](#)

Installation of GNOME Weather

Install GNOME Weather by running the following commands:

```
mkdir build &&
cd build &&
```

```
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: `gnome-weather`

Installed Libraries: None

Installed Directories: `/usr/share/org.gnome.Weather`

Short Descriptions

`gnome-weather` is a small application that allows you to monitor the current weather conditions anywhere in the world

Gucharmap-15.1.5

Introduction to Gucharmap

Gucharmap is a Unicode character map and font viewer. It allows you to browse through all the available Unicode characters and categories for the installed fonts, and to examine their detailed properties. It is an easy way to find the character you might only know by its Unicode name or code point.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.gnome.org/GNOME/gucharmap/-/archive/15.1.5/gucharmap-15.1.5.tar.bz2>
- Download MD5 sum: f50222e790637b951ae6a798d71b3f40
- Download size: 1.4 MB
- Estimated disk space required: 74 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Unicode Character Database: <https://www.unicode.org/Public/zipped/15.1.0/UCD.zip>
- CJK Unified Ideographs: <https://www.unicode.org/Public/zipped/15.1.0/Unihan.zip>

Gucharmap Dependencies

Required

[desktop-file-utils-0.27](#), [gsettings-desktop-schemas-46.1](#), [GTK+-3.24.43](#), [itstool-2.0.7](#), [pcre2-10.44](#), and [UnZip-6.0](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [Vala-0.56.17](#)

Optional

[appstream-glib-0.8.3](#) and [GTK-Doc-1.34.0](#)

Installation of Gucharmap

Note

This package uses two unversioned downloads. You may want to rename these to a versioned name, e.g. `UCD-15.zip` to make upgrades easier.

Install Gucharmap by running the following commands:

```
mkdir build      &&
cd build        &&
mkdir ucd       &&
pushd ucd       &&
  unzip ../../UCD.zip    &&
  cp -v ../../Unihan.zip .
popd            &&

meson setup --prefix=/usr      \
  --strip           \
  --buildtype=release \
  -D ucd_path=./ucd  \
  -D docs=false     \
  ..
ninja
```

This package does not have a test suite.

Now, as the `root` user:

```
rm -fv /usr/share/glib-2.0/schemas/org.gnome.Charmap.enums.xml &&
ninja install
```

Command Explanations

`--strip`: Although the package defaults to a release build without debug information, without this switch there will be some debug information in the binaries.

`-D ucd_path=./ucd`: This points to where the extracted main Unicode data files, and the zipped Unihan archive can be found.

`rm -fv /usr/share/glib-2.0/schemas/org.gnome.Charmap.enums.xml`: Old versions of this package installed an obsolete file that prevents the program from running, as if the main schema has not been installed. Forcibly removing works even if the old file is not present.

`-D docs=false`: This allows the package to build even if gtk-doc has not been installed. Remove this if you have installed that and wish to build the documentation.

`-D gir=false`: Use this if you have not installed [GLib-2.80.4](#) (with GObject Introspection).

`-D vapi=false`: Use this if you have not installed [Vala-0.56.17](#).

Contents

Installed Program: gucharmap

Installed Library: libgucharmap_2_90.so

Installed Directories: /usr/include/gucharmap-2.90

Short Descriptions

`gucharmap` is a Unicode character map and font viewer

`libgucharmap_2_90.so` contains the Gucharmap API functions

Seahorse-43.0

Introduction to Seahorse

Seahorse is a graphical interface for managing and using encryption keys. Currently it supports PGP keys (using GPG/GPGME) and SSH keys.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/seahorse/43/seahorse-43.0.tar.xz>
- Download MD5 sum: efa9fea2e1c4291c39d509eb366b9a56
- Download size: 1.3 MB
- Estimated disk space required: 43 MB
- Estimated build time: 0.1 SBU (Using parallelism=4)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/seahorse-43.0-upstream_fixes-1.patch

Seahorse Dependencies

Required

[Gcr-3.41.2](#), [GnuPG-2.4.5](#), [GPGME-1.23.2](#), [itstool-2.0.7](#), [libhandy-1.8.3](#), [libpwquality-1.4.5](#), [libsecret-0.21.4](#), and [Vala-0.56.17](#)

Recommended

[libsoup-3.4.4](#), [p11-kit-0.25.5](#), and [OpenSSH-9.8p1](#) (for managing SSH keys)

Optional

[Avahi-0.8](#)

Runtime Dependency

[gnome-keyring-46.2](#)

Installation of Seahorse

At first, apply a patch to build failures triggered by GnuPG-2.4 or newer, and GCC-14 or newer:

```
patch -Np1 -i ../seahorse-43.0-upstream_fixes-1.patch
```

Install Seahorse by running the following commands:

```
sed -i -r 's:(/apps):"/org/gnome\1:' data/*.xml &&
mkdir build &&
cd build &&
meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`sed ... data/*.xml`: This command fixes some deprecated entries in the schema templates.

Contents

Installed Program: seahorse

Installed Libraries: None

Installed Directories: /usr/{libexec,share,share/help/*}/seahorse

Short Descriptions

`seahorse` is the graphical interface for managing and using encryption keys

Snapshot-46.3

Introduction to Snapshot

The Snapshot package contains a program that takes pictures and videos from a Webcam.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/snapshot/46/snapshot-46.3.tar.xz>
- Download MD5 sum: 25f97e88005b933c382e5a0c2193599c
- Download size: 22 MB
- Estimated disk space required: 1.1 GB (2.9 MB installed)
- Estimated build time: 4.5 SBU

Snapshot Dependencies

Required

[libadwaita-1.5.3](#), [gst-plugins-bad-1.24.7](#), and [rustc-1.80.1](#)

Required at runtime

[pipewire-1.2.3](#)

Installation of Snapshot

Note

This package takes a long time to build because LTO is enabled, and the LTO pass is not parallelized.

Install Snapshot by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: snapshot

Installed Libraries: None

Installed Directories: /usr/share/snapshot

Short Descriptions

`snapshot` is a program used to take pictures and videos from a Webcam

Part IX. Xfce

Chapter 35. Xfce Desktop

Xfce is a desktop environment that aims to be fast and low on system resources, while still being visually appealing and user friendly.

Xfce embodies the traditional UNIX philosophy of modularity and re-usability. It consists of a number of components that provide the full functionality one can expect of a modern desktop environment. They are packaged separately and you can pick among the available packages to create the optimal personal working environment.

Build Xfce core packages in the order presented in the book for the easiest resolution of dependencies.

libxfce4util-4.18.2

Introduction to libxfce4util

The libxfce4util package is a basic utility library for the Xfce desktop environment.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/libxfce4util/4.18/libxfce4util-4.18.2.tar.bz2>
- Download MD5 sum: 9089e8974a581c5f39f610f2727c38ba
- Download size: 502 KB
- Estimated disk space required: 5.9 MB
- Estimated build time: less than 0.1 SBU

libxfce4util Dependencies

Required

[GLib-2.80.4](#) (with GObject Introspection)

Recommended

[Vala-0.56.17](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of libxfce4util

Install libxfce4util by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: xfce4-kiosk-query

Installed Library: libxfce4util.so

Installed Directories: /usr/include/xfce4 and /usr/share/gtk-doc/html/libxfce4util

Short Descriptions

xfce4-kiosk- query	Queries the given capabilities of <module> for the current user and reports whether the user has the capabilities or not. This tool is mainly meant for system administrators to test their Kiosk setup
libxfce4util.so	contains basic utility functions for the Xfce desktop environment

Xfconf-4.18.3

Introduction to Xfconf

Xfconf is the configuration storage system for Xfce.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfconf/4.18/xfconf-4.18.3.tar.bz2>
- Download MD5 sum: f807ed0a1b88af479ec70b28c1f78dcc
- Download size: 636 KB
- Estimated disk space required: 9.8 MB
- Estimated build time: 0.1 SBU

Xfconf Dependencies

Required

[libxfce4util-4.18.2](#)

Optional

[GTK-Doc-1.34.0](#) and [Vala-0.56.17](#)

Installation of Xfconf

Install Xfconf by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: xfconf-query

Installed Library: libxfconf-0.so

Installed Directories: /usr/include/xfce4/xfconf-0, /usr/lib/xfce4/xfconf, and /usr/share/gtk-doc/html/xfconf

Short Descriptions

xfconf-query	is a commandline utility to view or change any setting stored in Xfconf
libxfconf-0.so	contains basic functions for Xfce configuration

libxfce4ui-4.18.6

Introduction to libxfce4ui

The libxfce4ui package contains GTK+ 3 widgets that are used by other Xfce applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/libxfce4ui/4.18/libxfce4ui-4.18.6.tar.bz2>
- Download MD5 sum: 5da8e2ce2a154c9b3557e9b317c28ddb
- Download size: 885 KB
- Estimated disk space required: 14 MB
- Estimated build time: 0.1 SBU

libxfce4ui Dependencies

Required

[GTK+-3.24.43](#) and [Xfconf-4.18.3](#)

Recommended

[startup-notification-0.12](#)

Optional

[GTK-Doc-1.34.0](#), [libgtop-2.41.3](#), [libgudev-238](#), and [Glade](#)

Installation of libxfce4ui

Install libxfce4ui by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: None

Installed Libraries: libxfce4kbd-private-{2,3}.so and libxfce4ui-{1,2}.so

Installed Directories: /etc/xdg/xfce4, /usr/include/xfce4/libxfce4kbd-private-2, /usr/include/xfce4/libxfce4ui-{1,2}, and /usr/share/gtk-doc/html/libxfce4ui

Short Descriptions

libxfce4kbd-private-2.so is a private Xfce library for sharing code between Xfwm4 and Xfce4 Settings
libxfce4ui-1.so contains widgets that are used by other Xfce applications

Exo-4.18.0

Introduction to Exo

Exo is a support library used in the Xfce desktop. It also has some helper applications that are used throughout Xfce.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/exo/4.18/exo-4.18.0.tar.bz2>
- Download MD5 sum: 9d0be4e885eaf991e12dbc14d3fc628d
- Download size: 876 KB
- Estimated disk space required: 14 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Exo Dependencies

Required

[GTK+-3.24.43](#), [libxfce4ui-4.18.6](#), and [libxfce4util-4.18.2](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of Exo

Install Exo by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: exo-desktop-item-edit and exo-open

Installed Libraries: libexo-2.so

Installed Directories: /usr/include/exo-2 and /usr/share/gtk-doc/html/exo-2

Short Descriptions

<code>exo-desktop-item-edit</code>	is a command line utility to create or edit icons on the desktop
<code>exo-open</code>	is a command line frontend to the Xfce Preferred Applications framework. It can either be used to open a list of urls with the default URL handler or launch the preferred application for a certain category
<code>libexo-2.so</code>	contains additional widgets, a framework for editable toolbars, light-weight session management support and functions to automatically synchronise object properties (based on GObject Binding Properties)

Garcon-4.18.2

Introduction to Garcon

The Garcon package contains a freedesktop.org compliant menu implementation based on GLib and GIO.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/garcon/4.18/garcon-4.18.2.tar.bz2>
- Download MD5 sum: 153813ff2736f44fa7b6fa96068538d0
- Download size: 566 KB

- Estimated disk space required: 8.4 MB
- Estimated build time: 0.1 SBU

Garcon Dependencies

Required

[libxfce4ui-4.18.6](#) and [GTK+-3.24.43](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of Garcon

Install Garcon by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libgarcon-1.so and libgarcon-gtk3-1.so

Installed Directory: /usr/include/garcon-1, /usr/include/garcon-gtk3-1, and /usr/share/gtk-doc/html/garcon

Short Descriptions

<code>libgarcon-1.so</code>	contains functions that provide a freedesktop.org compliant menu implementation based on GLib and GIO
-----------------------------	---

libwnck-43.0

Introduction to libwnck

The libwnck package contains the Window Navigator Construction Kit.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/libwnck/43/libwnck-43.0.tar.xz>
- Download MD5 sum: cd21ef743a1e9286554401c5b28d5ec6
- Download size: 448 KB
- Estimated disk space required: 13 MB
- Estimated build time: less than 0.1 SBU

libwnck Dependencies

Required

[GTK+-3.24.43](#)

Recommended

[GLib-2.80.4](#) (with GObject Introspection) and [startup-notification-0.12](#)

Optional

Installation of libwnck

Install libwnck by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gtk_doc=true`: Use this option to build the API reference manual.

Contents

Installed Program: `wnckprop` and `wnck-urgency-monitor`

Installed Library: `libwnck-3.so`

Installed Directories: `/usr/include/libwnck-3.0` and `/usr/share/gtk-doc/html/libwnck-3.0`

Short Descriptions

`wnckprop` is used to print or modify the properties of a screen/workspace/window, or to interact with it
`libwnck-3.so` contains functions for writing pagers and task lists

xfce4-panel-4.18.6

Introduction to Xfce4 Panel

The Xfce4 Panel package contains the Xfce4 Panel.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfce4-panel/4.18/xfce4-panel-4.18.6.tar.bz2>
- Download MD5 sum: cadd05c820f0fbbae9d2acc5218fc7ed
- Download size: 1.6 MB
- Estimated disk space required: 38 MB
- Estimated build time: 0.3 SBU

Xfce4 Panel Dependencies

Required

[Cairo-1.18.0](#), [Exo-4.18.0](#), [Garcon-4.18.2](#), and [libwnck-43.0](#)

Optional

[GTK-Doc-1.34.0](#) and [libdbusmenu](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/xfce4-panel>

Installation of Xfce4 Panel

Install Xfce4 Panel by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-gtk-doc`: Use this parameter if GTK-Doc is installed and you wish to rebuild and install the API documentation.

Contents

Installed Programs: xfce4-panel, xfce4-popup-applicationsmenu, xfce4-popup-directorymenu, and xfce4-popup-windowmenu

Installed Library: libxfce4panel-2.0.so

Installed Directories: /etc/xdg/xfce4/panel, /usr/include/xfce4/libxfce4panel-1.0, /usr/lib/xfce4/panel, /usr/share/gtk-doc/html/libxfce4panel-1.0 and /usr/share/xfce4/panel

Short Descriptions

<code>xfce4-panel</code>	is the Xfce panel
<code>xfce4-popup-applicationsmenu</code>	is a shell script that uses D-Bus and Xfce Panel to display a popup menu of the installed applications
<code>xfce4-popup-directorymenu</code>	is a shell script that uses D-Bus and Xfce Panel to display a popup menu of your home folder and its subdirectories
<code>xfce4-popup-windowmenu</code>	is a shell script that uses DBus to display the Xfwm4 a popup menu
<code>libxfce4panel-2.0.so</code>	contains the Xfce Panel API functions

thunar-4.18.11

Introduction to thunar

Thunar is the Xfce file manager, a GTK+ 3 GUI to organise the files on your computer.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/thunar/4.18/thunar-4.18.11.tar.bz2>
- Download MD5 sum: 506e80f3fa94aca251b18c185b1303e8
- Download size: 2.8 MB
- Estimated disk space required: 67 MB
- Estimated build time: 0.4 SBU (Using parallelism=4)

Thunar Dependencies

Required

[Exo-4.18.0](#)

Required (Runtime)

[hicolor-icon-theme-0.18](#)

Recommended

[libgudev-238](#), [libnotify-0.8.3](#), and [pcre2-10.44](#)

Optional

[GTK-Doc-1.34.0](#) (for documentation), [Gvfs-1.54.2](#) (for remote browsing and automounting), [libexif-0.6.24](#) and [tumbler-4.18.2](#) (runtime)

Installation of Thunar

Install Thunar by running the following commands:

```
./configure --prefix=/usr \
--sysconfdir=/etc \
--docdir=/usr/share/doc/thunar-4.18.11 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: Thunar (symlink to thunar), thunar, and thunar-settings

Installed Library: libthunarl-3.so

Installed Directories: /etc/xdg/Thunar, /usr/include/thunarl-3, /usr/lib/Thunar, /usr/lib/thunarl-3, /usr/share/Thunar, /usr/share/doc/thunar-4.18.11, and /usr/share/gtk-doc/html/thunarl

Short Descriptions

<code>thunar</code>	is the Xfce file manager
<code>thunar-settings</code>	is a shell script that launches a dialog box to allow you to alter the behavior of Thunar
<code>libthunarl-3.so</code>	contains the Thunar extension library which permits adding new features to the Thunar file manager

thunarl-volman-4.18.0

Introduction to the Thunar Volume Manager

The Thunar Volume Manager is an extension for the Thunar file manager, which enables automatic management of removable drives and media.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/thunarl-volman/4.18/thunarl-volman-4.18.0.tar.bz2>
- Download MD5 sum: a0965931e78fe662ad134e63b1ab33b9
- Download size: 499 KB
- Estimated disk space required: 7.0 MB
- Estimated build time: less than 0.1 SBU

Thunarl Volume Manager Dependencies

Required

[Exo-4.18.0](#) and [libgudev-238](#)

Recommended

[libnotify-0.8.3](#)

Recommended Runtime Dependencies

[Gvfs-1.54.2](#)

Installation of the Thunar Volume Manager

Install the Thunar Volume Manager by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: thunar-volman and thunar-volman-settings

Installed Libraries: None

Installed Directories: None

Short Descriptions

`thunar-volman` is the Thunar Volume Manager, a command line utility to automatically mount or unmount removable media

`thunar-volman-settings` is a small GTK+ 3 application for changing Thunar Volume Manager settings

tumbler-4.18.2

Introduction to tumbler

The Tumbler package contains a D-Bus thumbnailing service based on the thumbnail management D-Bus specification. This is useful for generating thumbnail images of files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/tumbler/4.18/tumbler-4.18.2.tar.bz2>
- Download MD5 sum: 576ff73eb77530b001e36d565544a7f6
- Download size: 608 KB
- Estimated disk space required: 13 MB
- Estimated build time: 0.1 SBU

Tumbler Dependencies

Required

[GLib-2.80.4](#)

Optional

[cURL-8.9.1](#), [FreeType-2.13.3](#), [gdk-pixbuf-2.42.12](#), [gst-plugins-base-1.24.7](#), [GTK-Doc-1.34.0](#), [libjpeg-turbo-3.0.1](#), [libgsf-1.14.52](#), [libpng-1.6.43](#), [Poppler-24.08.0](#), [FFmpegThumbnailer](#), [libgepub](#), and [libopenraw](#)

Installation of Tumbler

Install Tumbler by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: tumblerd

Installed Library: libtumbler-1.so and several under /usr/lib/tumbler-1/plugins/

Installed Directories: /etc/xdg/tumbler, /usr/include/tumbler-1, /usr/lib/tumbler-1 and /usr/share/gtk-doc/html/tumbler

Short Descriptions

`tumblerd` is a D-Bus service for applications such as Thunar and Ristretto to use thumbnail images
`libtumbler-1.so` contains functions that the Tumbler daemon uses to create thumbnail images

xfce4-appfinder-4.18.1

Introduction to Xfce4 Appfinder

Xfce4 Appfinder is a tool to find and launch installed applications by searching the .desktop files installed on your system.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfce4-appfinder/4.18/xfce4-appfinder-4.18.1.tar.bz2>
- Download MD5 sum: 621d811f93b3edaac4a0205bc18ef5c5
- Download size: 552 KB
- Estimated disk space required: 7.1 MB
- Estimated build time: less than 0.1 SBU

Xfce4 Appfinder Dependencies

Required

[Garcon-4.18.2](#)

Installation of Xfce4 Appfinder

Install Xfce4 Appfinder by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: xfce4-appfinder and xfrun4 (symlink)

Installed Libraries: None

Installed Directories: None

Short Descriptions

`xfce4-appfinder` Is a GTK+ 3 application that enables you to quickly search through the .desktop files installed on your system looking for an application

xfce4-power-manager-4.18.4

Introduction to Xfce4 Power Manager

The Xfce4 Power Manager is a power manager for the Xfce desktop, Xfce power manager manages the power sources on the computer and the devices that can be controlled to reduce their power consumption (such as LCD brightness level or monitor sleep). In addition, Xfce4 Power Manager provides a set of freedesktop-compliant DBus interfaces to inform other applications about current power level so that they can adjust their power consumption.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfce4-power-manager/4.18/xfce4-power-manager-4.18.4.tar.bz2>
- Download MD5 sum: da60844080b55d3a9192e650f564c644
- Download size: 1.2 MB
- Estimated disk space required: 19 MB
- Estimated build time: 0.1 SBU

Xfce4 Power Manager Dependencies

Required

[libnotify-0.8.3](#), [UPower-1.90.4](#), and [xfce4-panel-4.18.6](#)

Recommended

[Polkit-125](#) (runtime, required for laptop backlight control)

Optional

[NetworkManager-1.48.8](#)

Installation of Xfce4 Power Manager

Install Xfce4 Power Manager by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: xfce4-power-manager, xfce4-power-manager-settings, xfce4-pm-helper, and xfpm-power-backlight-helper

Installed Libraries: libxfce4powermanager.so

Installed Directories: None

Short Descriptions

<code>xfce4-pm-helper</code>	is a helper program for the suspend and hibernate functions of <code>xfce4-power-manager</code>
<code>xfce4-power-manager</code>	is the Xfce Power Manager
<code>xfce4-power-manager-settings</code>	is a utility that comes with the Xfce Power Manager to access/change its configuration
<code>xfpm-power-backlight-helper</code>	is a command line utility to get or set the brightness of your screen

xfce4-settings-4.18.6

Introduction to Xfce4 Settings

The Xfce4 Settings package contains a collection of programs that are useful for adjusting your Xfce preferences.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfce4-settings/4.18/xfce4-settings-4.18.6.tar.bz2>
- Download MD5 sum: 37a5f463b2b81ac74a09edbda8ed4fb0
- Download size: 1.5 MB
- Estimated disk space required: 30 MB
- Estimated build time: 0.2 SBU

Xfce4 Settings Dependencies

Required

[Exo-4.18.0](#) and [Garcon-4.18.2](#)

Required (Runtime)

[gnome-icon-theme-3.12.0](#) or [lxde-icon-theme-0.5.1](#)

Recommended

[libcanberra-0.30](#), [libnotify-0.8.3](#) and [libxklavier-5.4](#)

Optional

[colord-1.4.7](#), [libinput-1.26.1](#), and [UPower-1.90.4](#)

Installation of Xfce4 Settings

Install Xfce4 Settings by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-sound-settings`: Use this switch to enable sound settings in GUI.

`--enable-pluggable-dialogs`: Use this switch to enable support for embedded settings dialogs.

Contents

Installed Programs: xfce4-accessibility-settings, xfce4-appearance-settings, xfce4-color-settings, xfce4-display-settings, xfce4-find-cursor, xfce4-keyboard-settings, xfce4-mime-helper, xfce4-mime-settings, xfce4-mouse-settings, xfce4-settings-editor, xfce4-settings-manager and xfsettingsd

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>xfce4-accessibility-settings</code>	is a GTK+ 3 GUI to allow you to change some of your keyboard and mouse preferences
<code>xfce4-appearance-settings</code>	is a GTK+ 3 GUI to allow you to change some of your theme, icon and font preferences
<code>xfce4-display-settings</code>	is a GTK+ 3 GUI to allow you to change some of your screen preferences
<code>xfce4-keyboard-settings</code>	is a GTK+ 3 GUI to allow you to change some of your keyboard preferences

<code>xfce4-mime-settings</code>	is a GTK+ 3 GUI to allow you to change which applications are used to handle different mime types
<code>xfce4-mouse-settings</code>	is a GTK+ 3 GUI to allow you to change some of your mouse preferences
<code>xfce4-settings-editor</code>	is a GTK+ 3 GUI to allow you to change your preferences stored in Xfconf
<code>xfce4-settings-manager</code>	is a GTK+ 3 GUI to allow you to change many of your Xfce preferences
<code>xfsettingsd</code>	is the Xfce settings daemon

Xfdesktop-4.18.1

Introduction to Xfdesktop

Xfdesktop is a desktop manager for the Xfce Desktop Environment. Xfdesktop sets the background image / color, creates the right click menu and window list and displays the file icons on the desktop using Thunar libraries.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfdesktop/4.18/xfdesktop-4.18.1.tar.bz2>
- Download MD5 sum: e675c2989436dc724fef402d1db0125d
- Download size: 2.1 MB
- Estimated disk space required: 21 MB
- Estimated build time: 0.1 SBU

Xfdesktop Dependencies

Required

[Exo-4.18.0](#) and [libwnck-43.0](#)

Recommended

[libnotify-0.8.3](#), [startup-notification-0.12](#) and [thunar-4.18.11](#)

Installation of Xfdesktop

Install Xfdesktop by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: xfdesktop and xfdesktop-settings

Installed Libraries: None

Installed Directories: /usr/share/backgrounds/xfce and /usr/share/pixmaps/xfdesktop

Short Descriptions

<code>xfdesktop</code>	is the Xfce Desktop Environment's desktop manager
<code>xfdesktop-settings</code>	is a GTK+ 3 application that allows you to change your desktop background, some preferences for the right click menu and what icons are displayed on the desktop

Xfwm4-4.18.0

Introduction to Xfwm4

Xfwm4 is the window manager for Xfce.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfwm4/4.18/xfwm4-4.18.0.tar.bz2>
- Download MD5 sum: 181415e457c86124fa5f8aa5d715b967
- Download size: 1.2 MB
- Estimated disk space required: 27 MB
- Estimated build time: 0.2 SBU

Xfwm4 Dependencies

Required

[libwnck-43.0](#) and [libxfc4ui-4.18.6](#)

Recommended

[startup-notification-0.12](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/xfwm4>

Installation of Xfwm4

Install Xfwm4 by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: xfwm4, xfwm4-settings, xfwm4-tweaks-settings and xfwm4-workspace-settings

Installed Libraries: None

Installed Directories: /usr/lib/xfce4/xfwm4, /usr/share/themes/{Daloa,Default{-hdpi,-xhdpi},Kokodi,Moheli}, and /usr/share/xfwm4

Short Descriptions

<code>xfwm4</code>	is the Xfce window manager
<code>xfwm4-settings</code>	is a GTK+ 3 application that allows setting some preferences such as your theme, keyboard shortcuts and mouse focus behavior
<code>xfwm4-tweaks-settings</code>	is a GTK+ 3 application that allows setting some more preferences for Xfwm4
<code>xfwm4-workspace-settings</code>	is a GTK+ 3 application that allows setting your workspace preferences

xfce4-session-4.18.4

Introduction to Xfce4 Session

Xfce4 Session is a session manager for Xfce. Its task is to save the state of your desktop (opened applications and their location) and restore it during a next startup. You can create several different sessions and choose one of them on startup.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/xfce/xfce4-session/4.18/xfce4-session-4.18.4.tar.bz2>
- Download MD5 sum: 620c369a44c6b866c80d31e4685d6bef
- Download size: 901 KB
- Estimated disk space required: 16 MB
- Estimated build time: 0.1 SBU

Xfce4 Session Dependencies

Required

[libwnck-43.0](#) and [libxfce4ui-4.18.6](#)

Recommended

[desktop-file-utils-0.27](#), [xfce4-screensaver](#) or [XScreenSaver-6.09](#), [shared-mime-info-2.4](#) and [polkit-gnome-0.105](#)

Required Runtime

[Xfdesktop-4.18.1](#)

Installation of Xfce4 Session

Install Xfce4 Session by running the following commands:

```
./configure --prefix=/usr \
--sysconfdir=/etc \
--disable-legacy-sm &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-legacy-sm`: This switch disables legacy session management which isn't necessary on modern system.

Configuring Xfce4 Session

There are several optional run time dependencies for Xfce4: [dbus-1.14.10](#), [GnuPG-2.4.5](#), [hicolor-icon-theme-0.18](#), [OpenSSH-9.8p1](#), and [xinit-1.4.2](#)

When building an Xfce4 package, some configuration files are installed in `/usr/share/applications`, `/usr/share/icons`, and `/usr/share/mime`. In order to use those files in your Xfce4 session, you need to update various databases. Do that by running, as the `root` user (you need to have the recommended dependencies installed):

```
update-desktop-database &&
update-mime-database /usr/share/mime
```

Starting Xfce4

You can start Xfce4 from a TTY using [xinit-1.4.2](#), or from a graphical display manager, such as [lightdm-1.32.0](#).

To start Xfce4 using [xinit-1.4.2](#), run the following commands:

```
cat > ~/.xinitrc << "EOF"
dbus-launch --exit-with-x11 startxfce4
EOF

startx
```

The X session starts on the first unused virtual terminal, normally vt7. You can switch to another vtn simultaneously pressing the keys Ctrl-Alt-Fn ($n=1, 2, \dots$). To switch back to the X session, normally started at vt7, use Ctrl-Alt-F7. The vt where the command `startx` was executed will display many messages, including X starting messages, applications

automatically started with the session, and eventually, some warning and error messages. You may prefer to redirect those messages to a log file, which not only will keep the initial vt uncluttered, but can also be used for debugging purposes. This can be done starting X with:

```
startx &> ~/x-session-errors
```

When shutting down or rebooting, the shutdown messages appear on the vt where X was running. If you wish to see those messages, simultaneously press keys Alt-F7 (assuming that X was running on vt7).

Contents

Installed Programs: startxfce4, xfce4-session, xfce4-session-logout, xfce4-session-settings, and xflock4

Installed Library: None

Installed Directories: /usr/lib/xfce4/session

Short Descriptions

startxfce4	is a script responsible for starting an Xfce session.
xfce4-session	starts up the Xfce Desktop Environment
xfce4-session-logout	logs out from Xfce
xfce4-session-settings	is a GTK+ 3 GUI which allows you to alter your preferences for your Xfce Session

Chapter 36. Xfce Applications

This is a small collection of optional applications that add extra capabilities to your Xfce desktop.

Parole-4.18.1

Introduction to Parole

Parole is a DVD/CD/music player for Xfce that uses GStreamer.

This package is known to build and work properly using an LFS 12.2 platform.

Note

Although this version of parole works with most files, it is unable to play DVDs, reporting an error in the gstreamer backend.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/apps/parole/4.18/parole-4.18.1.tar.bz2>
- Download MD5 sum: 29a409b4b22c2d91f210679e5708a19d
- Download size: 916 KB
- Estimated disk space required: 19 MB
- Estimated build time: 0.2 SBU

Parole Dependencies

Required

[dbus-glib-0.112](#), [gst-plugins-base-1.24.7](#), [gst-plugins-good-1.24.7](#), and [libxfce4ui-4.18.6](#)

Recommended

[libnotify-0.8.3](#) and [taglib-2.0.1](#)

Optional

[GTK-Doc-1.34.0](#)

Installation of Parole

Install Parole by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Using Parole

If you have installed Gstreamer Plugins Ugly with support for libdvdnav and libdvdread and would like to use Parole to play a DVD, click Media> Open location and enter dvd:// into the box.

Similarly, to play a CD, click Media> Open location and enter cdda:// into the box.

Contents

Installed Program: parole

Installed Libraries: Two libraries under /usr/lib/parole-0/

Installed Directories: /usr/include/parole, /usr/lib/parole-0 and /usr/share/parole

Short Descriptions

`parole` is a GTK+ 3 media player that uses GStreamer

xfce4-terminal-1.1.3

Introduction to Xfce4 Terminal

Xfce4 Terminal is a GTK+3 terminal emulator. This is useful for running commands or programs in the comfort of an Xorg window; you can drag and drop files into the Xfce4 Terminal or copy and paste text with your mouse.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/apps/xfce4-terminal/1.1/xfce4-terminal-1.1.3.tar.bz2>
- Download MD5 sum: 8c986841d70c7480e8bd928466b8cdc2
- Download size: 1.0 MB
- Estimated disk space required: 15 MB
- Estimated build time: 0.1 SBU

Xfce4 Terminal Dependencies

Required

[libxfce4ui-4.18.6](#) and [VTE-0.76.4](#)

Installation of Xfce4 Terminal

Install Xfce4 Terminal by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program:	xfce4-terminal
Installed Libraries:	None
Installed Directory:	/usr/share/xfce4/terminal

Short Descriptions

`xfce4-terminal` is a GTK+ 3 terminal emulator

Xfburn-0.7.2

Introduction to Xfburn

Xfburn is a GTK+ 3 GUI frontend for Libisoburn. This is useful for creating CDs and DVDs from files on your computer or ISO images downloaded from elsewhere.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/apps/xfburn/0.7/xfburn-0.7.2.tar.bz2>
- Download MD5 sum: a895d4d02cf2ee794162815c3f531a42
- Download size: 1.2 MB
- Estimated disk space required: 15 MB
- Estimated build time: 0.1 SBU

Xfburn Dependencies

Required

`Exo-4.18.0`, `libburn-1.5.6`, `libisofs-1.5.6`, and `libxfc4ui-4.18.6`

Optional

`gst-plugins-base-1.24.7` and `Cdrdao-1.2.4` (runtime)

Installation of Xfburn

Install Xfburn by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program:	xfburn
Installed Libraries:	None
Installed Directory:	/usr/share/xfburn

Short Descriptions

`xfburn` is a GTK+ 3 application for creating CDs and DVDs

Ristretto-0.13.2

Introduction to Ristretto

Ristretto is a fast and lightweight image viewer for the Xfce desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/apps/ristretto/0.13/ristretto-0.13.2.tar.bz2>
- Download MD5 sum: f3b1706a991a34f9866b0b2c27c6d0e1
- Download size: 823 KB
- Estimated disk space required: 9.9 MB
- Estimated build time: 0.1 SBU

Ristretto Dependencies

Required

[libexif-0.6.24](#) and [libxfce4ui-4.18.6](#)

Optional

[tumbler-4.18.2](#) (runtime)

Installation of Ristretto

Install Ristretto by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: ristretto

Installed Libraries: None

Installed Directories: None

Short Descriptions

`ristretto` is a fast and lightweight image viewer

xfce4-dev-tools-4.18.1

Introduction to the Xfce4 Development Tools

The Xfce4 Development Tools are a collection of tools and macros for building some Xfce applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://archive.xfce.org/src/xfce/xfce4-dev-tools/4.18/xfce4-dev-tools-4.18.1.tar.bz2>
- Download MD5 sum: 69b4cd255a0b8f12bbdc9b10c433b223
- Download size: 356 KB
- Estimated disk space required: 2.6 MB

- Estimated build time: less than 0.1 SBU

The Xfce4 Development Tools Dependencies

Required

[GLib-2.80.4](#)

Installation of the Xfce4 Development Tools

Install the Xfce4 Development Tools by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: xdt-autogen and xdt-csource

Installed Libraries: None

Installed Directories: None

xfce4-notifyd-0.9.6

Introduction to the Xfce4 Notification Daemon

The Xfce4 Notification Daemon is a small program that implements the "server-side" portion of the Freedesktop desktop notifications specification. Applications that wish to pop up a notification bubble in a standard way can use Xfce4-Notifyd to do so by sending standard messages over D-Bus using the org.freedesktop.Notifications interface.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/apps/xfce4-notifyd/0.9/xfce4-notifyd-0.9.6.tar.bz2>
- Download MD5 sum: 1640c56ec1062dc46b5691eab12db9b6
- Download size: 728 KB
- Estimated disk space required: 11 MB
- Estimated build time: 0.1 SBU

The Xfce4 Notification Daemon Dependencies

Required

[libnotify-0.8.3](#), [libxfce4ui-4.18.6](#), [libcanberra-0.30](#), [xfce4-dev-tools-4.18.1](#), and [xfce4-panel-4.18.6](#)

Installation of the Xfce4 Notification Daemon

Install the Xfce4 Notification Daemon by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

You can test the notification daemon with the command `notify-send`, to be run in a graphical environment:

```
notify-send -i info Information "Hi ${USER}, This is a Test"
```

Contents

Installed Program: xfce4-notifyd-config

Installed Libraries: libnotification-plugin.so

Installed Directories: /usr/share/themes/Bright/xfce-notify-4.0, /usr/share/themes/Default/xfce-notify-4.0, /usr/share/themes/Retro/xfce-notify-4.0, /usr/share/themes/Smoke/xfce-notify-4.0 and /usr/share/themes/ZOMG-PONIES!/xfce-notify-4.0

Short Descriptions

`xfce4-notifyd-`
`config` is a GTK+ GUI that allows you to change some of your preferences (theme and screen position) for the notifications that the Xfce4 Notification Daemon displays

xfce4-pulseaudio-plugin-0.4.8

Introduction to the Xfce4 Pulseaudio Plugin

The Xfce4 Pulseaudio Plugin is a plugin for the Xfce panel which provides a convenient way to adjust the audio volume of the PulseAudio sound system and to an auto mixer tool like pavucontrol. It can optionally handle multimedia keys for controlling the audio volume.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.xfce.org/src/panel-plugins/xfce4-pulseaudio-plugin/0.4/xfce4-pulseaudio-plugin-0.4.8.tar.bz2>
- Download MD5 sum: 83317e672e843e0b14d0da24ed301b3b
- Download size: 428 KB
- Estimated disk space required: 6.4 MB
- Estimated build time: less than 0.1 SBU

Xfce4 Pulseaudio Plugin Dependencies

Required

[libnotify-0.8.3](#), [PulseAudio-17.0](#), and [xfce4-panel-4.18.6](#)

Recommended

[keybinder-3.0-0.3.2](#) and [pavucontrol-6.1](#) (runtime)

Installation of the Xfce4 Pulseaudio Plugin

Install the Xfce4 Pulseaudio Plugin by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: None

Installed Libraries: libpulseaudio-plugin.so

Installed Directories: None

Part X. LXQt

Chapter 37. LXQt Desktop

LXQt is an extremely fast-performing and energy-saving desktop environment.

LXQt is a Qt based Desktop Environment. It is focused on being a classic desktop with a modern look and feel. It can customize appearance everywhere. The panel(s) have plugins and settings. It is Window Manager agnostic, has multi-language support and standard keyboard shortcuts. LXQt uses less CPU and less RAM than most other environments. It is especially useful for cloud computers with low hardware specifications, such as netbooks, mobile devices (e.g. MIDs) or older computers, but can be used with modern hardware.

Build LXQt core packages in the order presented in the book for the easiest resolution of dependencies.

LXQt Desktop Pre-Install Instructions

Notes about building LXQt dependencies

To build LXQt, the following steps are recommended:

1. First, build the whole [Xorg Chapter](#), including Wayland support with required and recommended dependencies. Test that the environment works with twm as the window manager. With experience, some of the packages can be omitted, but that is not recommended for first time users.
2. Second, build the required parts of Qt. The full package, [Qt-6.7.2](#) is quite long and has a lot of dependencies.
3. Finally build the LXQt desktop using the pages below in order. The packages in the applications section are optional, but as a minimum, [qterminal-2.0.1](#) is recommended.

KF5, Plasma, and Misc Dependencies

kwindowsystem-6.5.0 for lxqt

Introduction to kwindowsystem

The kwindowsystem provides information about, and allows interaction with, the windowing system. It provides a high level API that is windowing system independent and has platform specific implementations.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the KF6 set of packages. If [KDE Frameworks-6.5.0](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/kwindowsystem-6.5.0.tar.xz>
- Download MD5 sum: 5631073e18780513a3005159dd73e290
- Download size: 2.3 MB
- Estimated disk space required: 73 MB
- Estimated build time: 0.7 SBU

kwindowsystem Dependencies

Required

[extra-cmake-modules-6.5.0](#), [plasma-wayland-protocols-1.13.0](#), [Qt-6.7.2](#), and [Xorg Libraries](#)

Installation of kwindowsystem

Install kwindowsystem by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libKF5WindowSystem.so

Installed Directories: \$QT6DIR/qml/org/kde, \$QT6DIR/plugins/kf6, /usr/share/qlogging-categories6, /usr/lib/cmake/KF5WindowSystem, and /usr/include/KF6

Short Descriptions

`libKF5WindowSystem.so` contains the KF5 Windowing API functions

kconfig-6.5.0 for lxqt

Introduction to kconfig

The kconfig package provides access to configuration files.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the KF6 set of packages. If [KDE Frameworks-6.5.0](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/kconfig-6.5.0.tar.xz>
- Download MD5 sum: 698ab95acdc713571f917bb70073e2df
- Download size: 336 KB
- Estimated disk space required: 25 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

kconfig Dependencies

Required

[extra-cmake-modules-6.5.0](#) and [Qt-6.7.2](#)

Installation of kconfig

Install kconfig by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_TESTING=OFF \
      -W no-dev .. &&
make
```

```
-D CMAKE_INSTALL_LIBEXECDIR=libexec \
-D KDE_INSTALL_USE_QT_SYS_PATHS=ON \
-D BUILD_TESTING=OFF \
-W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: kreadconfig6 and kwritedconfig6

Installed Library: libKF6ConfigCore.so, libKF6ConfigGui.so, and libKF6ConfigQml.so

Installed Directories: /usr/lib/libexec/kf6, /usr/lib/qml/org/kde/config, /usr/lib/cmake/KF6Config, and /usr/include/KF6/{KConfigCore,KConfigQml,KConfigGui}

Short Descriptions

<code>kreadconfig6</code>	reads a configuration file using the KConfig interface
<code>kwritedconfig6</code>	writes to a configuration file using the KConfig interface
<code>libKF6ConfigCore.so</code>	contains the core configuration library for KF6
<code>libKF6ConfigGui.so</code>	contains functions for a GUI for KConfig
<code>libKF6ConfigQml.so</code>	contains a QML interface to KConfig

solid-6.5.0 for lxqt

Introduction to solid

Solid is a device integration framework. It provides a way of querying and interacting with hardware independently of the underlying operating system.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the KF6 set of packages. If [KDE Frameworks-6.5.0](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/solid-6.5.0.tar.xz>
- Download MD5 sum: 712a3c59785bd0de4862cce34de58e4c
- Download size: 308 KB
- Estimated disk space required: 30 MB
- Estimated build time: 0.4 SBU (Using parallelism=4)

***solid* Dependencies**

Required

[extra-cmake-modules-6.5.0](#) and [Qt-6.7.2](#)

Optional (runtime)

[UDisks-2.10.1](#), [UPower-1.90.4](#), [libimobiledevice](#), and [media-player-info](#)

Installation of solid

Install solid by running the following commands:

```

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr          \
      -D CMAKE_BUILD_TYPE=Release          \
      -D CMAKE_INSTALL_LIBEXECDIR=libexec \
      -D KDE_INSTALL_USE_QT_SYS_PATHS=ON   \
      -D BUILD_TESTING=OFF                \
      -W no-dev ..                      &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: solid-hardware6

Installed Library: libKF6Solid.so

Installed Directories: /usr/include/KF6/Solid and /usr/lib/cmake/KF6Solid

Short Descriptions

`solid-hardware6` queries hardware information from the system

`libKF6Solid.so` provides a way of querying and interacting with hardware independently of the underlying operating system

kidletime-6.5.0 for lxqt

Introduction to kidletime

KIdleTime is used to report the idle time of users and the system. It is useful not only for determining the current idle time of the PC, but also for getting notified upon idle time events, such as custom timeouts or user activity.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the KF6 set of packages. If [KDE Frameworks-6.5.0](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <https://download.kde.org/stable/frameworks/6.5/kidletime-6.5.0.tar.xz>
- Download MD5 sum: fb7a5aec22e99bc69d4c4d50ee418dd5
- Download size: 32 KB
- Estimated disk space required: 4.9 MB
- Estimated build time: 0.2 SBU

kidletime Dependencies

Required

[extra-cmake-modules-6.5.0](#), [plasma-wayland-protocols-1.13.0](#), and [Qt-6.7.2](#)

Installation of kidletime

Install kidletime by running the following commands:

```

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr          \
      -D CMAKE_BUILD_TYPE=Release          \
      -D CMAKE_INSTALL_LIBEXECDIR=libexec \
      -D KDE_INSTALL_USE_QT_SYS_PATHS=ON   \
      -D BUILD_TESTING=OFF                \
      -W no-dev ..                      &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libKF6IdleTime.so

Installed Directories: /usr/lib/plugins/kf6/org.kde.kidletime.platforms, /usr/lib/cmake/KF6IdleTime, and /usr/include/KF6/KIdleTime

Short Descriptions

`libKF6IdleTime.so` contains the KDE API for idle time reports

kwayland-6.1.4 for Ixqt

Introduction to kwayland

kwayland is a Qt-style API to interact with the wayland-client and wayland-server API.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the KF6 set of packages. If [KDE Frameworks-6.5.0](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <http://download.kde.org/stable/plasma/6.1.4/kwayland-6.1.4.tar.xz>
- Download MD5 sum: d297f096ac9492262b8274a024618bd3
- Download size: 132 KB
- Estimated disk space required: 13 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

kwayland Dependencies

Required

[extra-cmake-modules-6.5.0](#), [Mesa-24.1.5](#) (built with [Wayland-1.23.0](#) support), [plasma-wayland-protocols-1.13.0](#), and [Qt-6.7.2](#)

Installation of kwayland

Install kwayland by running the following commands:

```

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr          \
      -D CMAKE_BUILD_TYPE=Release          \
      -D CMAKE_INSTALL_LIBEXECDIR=libexec \
      -D KDE_INSTALL_USE_QT_SYS_PATHS=ON   \
      -D BUILD_TESTING=OFF                \
      -W no-dev ..                      &&
make

```

```
-D CMAKE_BUILD_TYPE=Release      \
-D CMAKE_INSTALL_LIBEXECDIR=libexec \
-D KDE_INSTALL_USE_QT_SYS_PATHS=ON \
-D BUILD_TESTING=OFF           \
-W no-dev ..                   &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libKWaylandClient.so

Installed Directories: /usr/include/KWayland, and /usr/lib/cmake/KWayland

Short Descriptions

`libKWaylandClient.so` contains the wayland-client Qt-style wrapper library

libkscreen-6.1.4 for Ixqt

Introduction to libkscreen

The libkscreen package contains the KDE Screen Management library.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the plasma set of packages. If [Plasma-6.1.4](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <http://download.kde.org/stable/plasma/6.1.4/libkscreen-6.1.4.tar.xz>
- Download MD5 sum: dfd2bd982b5ab4daa5a3545e54cf9cb3
- Download size: 120 KB
- Estimated disk space required: 17 MB
- Estimated build time: 0.3 SBU (using parallelism=4)

libkscreen Dependencies

Required

[plasma-wayland-protocols-1.13.0](#) and [Qt-6.7.2](#)

Installation of libkscreen

Install libkscreen by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_BUILD_TYPE=Release           \
-D CMAKE_INSTALL_LIBEXECDIR=libexec \
-D KDE_INSTALL_USE_QT_SYS_PATHS=ON   \
-D BUILD_TESTING=OFF                 \
-W no-dev ..                         &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `kscreen-doctor`

Installed Library: `libKF6Screen.so` and `libKF6ScreenDpms.so`

Installed Directories: /usr/lib/cmake/KF6Screen, /usr/include/KF6/KScreen, and /usr/lib/plugins/kf6/kscreen

Short Descriptions

<code>kscreen-doctor</code>	allows modifying the screen setup from the command line
<code>libKF6Screen.so</code>	contains the KDE Screen Management library
<code>libKF6ScreenDpms.so</code>	contains API functions for handling DPMS

layer-shell-qt-6.1.4 for Ixqt

Introduction to layer-shell-qt

This component is meant for applications to be able to easily use clients based on a "wtl-layer-shell" protocol. Clients can use this interface to assign the `surface_layer` role to `wl_surfaces`. Such surfaces are assigned to a "layer" of the output and rendered with a defined z-depth respective to each other.

This package is known to build and work properly using an LFS 12.2 platform.

Important

This package is extracted from the plasma set of packages. If [Plasma-6.1.4](#) is built, do **NOT** also build this package as presented here.

Package Information

- Download (HTTP): <http://download.kde.org/stable/plasma/6.1.4/layer-shell-qt-6.1.4.tar.xz>
- Download MD5 sum: 2e3a5220acf76e2c0edf9701978e802b
- Download size: 36 KB
- Estimated disk space required: 4.7 MB
- Estimated build time: 0.2 SBU (using parallelism=4)

layer-shell-qt Dependencies

Required

[extra-cmake-modules-6.5.0](#) and [Qt-6.7.2](#)

Installation of layer-shell

Install layer-shell-qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libLayerShellQtInterface.so

Installed Directories: /usr/lib/qml/org/kde/layershell, /usr/include/LayerShellQt, and /usr/lib/wayland-shell-integration

muparser-2.3.4

Introduction to muparser

The muparser package is a fast math parser library for C and C++.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/beltoforion/muparser/archive/v2.3.4/muparser-2.3.4.tar.gz>
- Download MD5 sum: b92180a648be88238008ea01a597ccb9
- Download size: 112 KB
- Estimated disk space required: 5.1 MB
- Estimated build time: 0.1 SBU

muparser Dependencies

Required

[CMake-3.30.2](#)

Installation of muparser

Install muparser by running the following commands:

```
mkdir -v build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libmuparser.so

Installed Directories: /usr/lib/cmake/muparser

Short Descriptions

`muparser.so` provides a fast math parser library for C and C++

LXQt Desktop Components

lxqt-build-tools-2.0.0

Introduction to lxqt-build-tools

The lxqt-build-tools package provides several tools needed to build LXQt itself as well as other components maintained by the LXQt project.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-build-tools/releases/download/2.0.0/lxqt-build-tools-2.0.0.tar.xz>
- Download MD5 sum: f6b52c3a792dd48d7debfa90eed52f3b
- Download size: 28 KB
- Estimated disk space required: 1.0 MB
- Estimated build time: less than 0.1 SBU

lxqt-build-tools Dependencies

Required

[CMake-3.30.2](#) and [Qt-6.7.2](#)

Installation of lxqt-build-tools

Install lxqt-build-tools by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      .. &&

make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: lxqt2-transupdate

Installed Library: None

Installed Directories: /usr/share/cmake/lxqt2-build-tools

Short Descriptions

`lxqt2-transupdate` updates translations for LXQt components

libqtxdg-4.0.0

Introduction to libqtxdg

The libqtxdg is a Qt implementation of the freedesktop.org xdg specifications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/libqtxdg/releases/download/4.0.0/libqtxdg-4.0.0.tar.xz>
- Download MD5 sum: 4aaf75449cb6c8251f3b2fc21e4c5122
- Download size: 72 KB
- Estimated disk space required: 9.5 MB

- Estimated build time: 0.3 SBU

libqtxdg Dependencies

Required

[CMake-3.30.2](#) and [Qt-6.7.2](#)

Optional (runtime)

[GTK+-3.24.43](#) (for `gtk-update-icon-cache`)

Installation of libqtxdg

Install libqtxdg by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&

make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libQt6Xdg.so and libQt6XdgIconLoader.so

Installed Directories: /usr/include/{qt6xdg,qt6xdgiconloader} and /usr/share/cmake/{qt6xdg,qt6xdgiconloader}

Short Descriptions

<code>libQt6Xdg.so</code>	contains the libQt6Xdg API functions
<code>libQt6XdgIconLoader.so</code>	contains the libQt6XdgIconLoader API functions

Ixqt-menu-data-2.0.0

Introduction to Ixqt-menu-data

The Ixqt-menu-data package contains compliant menu files for LXQt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Ixqt/Ixqt-menu-data/releases/download/2.0.0/Ixqt-menu-data-2.0.0.tar.xz>
- Download MD5 sum: ba924239447b2d5d19c3024871fba199
- Download size: 56 KB
- Estimated disk space required: 1.8 MB
- Estimated build time: less than 0.1 SBU

Ixqt-menu-data Dependencies

Required

[Ixqt-build-tools-2.0.0](#)

Installation of lxqt-menu-data

Install lxqt-menu-data by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: None

Installed Library: None

Installed Directories: /etc/xdg/menus, /usr/share/desktop-directories, and /usr/share/cmake/lxqt-menu-data

liblxqt-2.0.0

Introduction to liblxqt

The liblxqt is the core utility library for all LXQt components.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/liblxqt/releases/download/2.0.0/liblxqt-2.0.0.tar.xz>
- Download MD5 sum: c1c51cb0d759cc6beb8b576b5a2c8285
- Download size: 84 KB
- Estimated disk space required: 7.8 MB
- Estimated build time: 0.3 SBU

liblxqt Dependencies

Required

[libqtxdg-4.0.0](#), [kwindowsystem-6.5.0 for lxqt](#), and [polkit-qt-0.200.0](#)

Installation of liblxqt

Install liblxqt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: lxqt-backlight_backend

Installed Library: liblxqt.so

Installed Directories: /usr/include/lxqt and /usr/share/cmake/lxqt

Short Descriptions

lxqt-backlight_backend	sets the backlight level for a display
liblxqt.so	contains the liblxqt API functions

libsystat-1.0.0

Introduction to libsysstat

The libsysstat package contains a library used to query system information and statistics.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/libsysstat/releases/download/1.0.0/libsysstat-1.0.0.tar.xz>
- Download MD5 sum: 8a48fa8de906f9e73bd9ef91e39128eb
- Download size: 20 KB
- Estimated disk space required: 2.1 MB
- Estimated build time: less than 0.1 SBU

libsystat Dependencies

Required

[lxqt-build-tools-2.0.0](#)

Installation of libsysstat

Install libsysstat by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ...
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: None

Installed Library: libsysstat-qt6.so

Installed Directories: /usr/include/sysstat-qt6 and /usr/share/cmake/sysstat-qt6

Short Descriptions

libsystat-qt6.so	contains the libsysstat-qt6 API functions
------------------	---

qtxdg-tools-4.0.0

Introduction to qtxdg-tools

The qtxdg-tools contains a CLI MIME tool for handling file associations and opening files with their default applications. This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/qtxdg-tools/releases/download/4.0.0/qtxdg-tools-4.0.0.tar.xz>
- Download MD5 sum: 6b9022f7f39f8bff398e69206ad01a93
- Download size: 20 KB
- Estimated disk space required: 1.8 MB
- Estimated build time: 0.1 SBU

qtxdg-tools Dependencies

Required

[libqtxdg-4.0.0](#)

Installation of qtxdg-tools

Install qtxdg-tools by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: qtxdg-mat

Installed Library: None

Installed Directories: /usr/share/cmake/qtxdg-tools

Short Descriptions

<code>qtxdg-mat</code>	queries and modifies defaults for applications, such as the default terminal, web browser, email client, and file manager
------------------------	---

libfm-extra-1.3.2

Introduction to libfm-extra

The libfm-extra package contains a library and other files required by the `menu-cache-gen` program in /usr/bin/libexec installed by [menu-cache-1.1.0](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/pcmancfm/libfm-1.3.2.tar.xz>
- Download MD5 sum: c87a0ff41ae77825079b2f785ec0741e
- Download size: 924 KB
- Estimated disk space required: 11 MB
- Estimated build time: less than 0.1 SBU

libfm-extra Dependencies

Required

[GLib-2.80.4](#)

Installation of libfm-extra

Install libfm-extra by running the following commands:

```
./configure --prefix=/usr      \
            --sysconfdir=/etc \
            --with-extra-only \
            --with-gtk=no \
            --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--with-extra-only`: This switch disables all components except for the libfm-extra library.

`--with-gtk=no`: This switch disables support for GTK+ because it is not necessary for this package.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libfm-extra.so

Installed Directories: /usr/include/libfm (symlink) and /usr/include/libfm-1.0

Short Descriptions

`libfm-extra.so` contains the libfm-extra API functions

menu-cache-1.1.0

Introduction to Menu Cache

The Menu Cache package contains a library for creating and utilizing caches to speed up the manipulation for freedesktop.org defined application menus.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/lxde/menu-cache-1.1.0.tar.xz>
- Download MD5 sum: 99999a0bca48b980105208760c8fd893
- Download size: 260 KB
- Estimated disk space required: 3.7 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/menu-cache-1.1.0-consolidated_fixes-1.patch

Menu Cache Dependencies

Required

[libfm-extra-1.3.2](#)

Optional

[GTK-Doc-1.34.0](#) and [xdg-utils-1.2.1](#)

Installation of Menu Cache

First fix a memory leak and “multiple definitions” issues:

```
patch -Np1 -i ../menu-cache-1.1.0-consolidated_fixes-1.patch
```

Install Menu Cache by running the following commands:

```
./configure --prefix=/usr \
            --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-gtk-doc`: Use this option if GTK-Doc is installed and you wish to build and install the API documentation.

Contents

Installed Programs: None

Installed Library: libmenu-cache.so

Installed Directories: /usr/{include,libexec,share/gtk-doc/html}/menu-cache

Short Descriptions

`libmenu-cache.so` contains the menu-cache API functions

libfm-qt-2.0.2

Introduction to libfm-qt

libfm-qt is the Qt port of libfm, a library providing components to build desktop file managers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/libfm-qt/releases/download/2.0.2/libfm-qt-2.0.2.tar.xz>
- Download MD5 sum: cc0305facff6c194f3d1663924ddc89e
- Download size: 404 KB
- Estimated disk space required: 27 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

libfm-qt Dependencies

Required

[CMake-3.30.2](#), [libexif-0.6.24](#), [menu-cache-1.1.0](#), and [Qt-6.7.2](#)

Installation of libfm-qt

Install libfm-qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libfm-qt6.so

Installed Directories: /usr/include/libfm-qt6, /usr/share/libfm-qt6, and /usr/share/cmake/fm-qt6

Short Descriptions

libfm-qt6.so contains the libfm-qt6 API functions

Ixqt-globalkeys-2.0.0

Introduction to Ixqt-globalkeys

The Ixqt-globalkeys package contains a daemon used to register global keyboard shortcuts as well as an editor for keyboard shortcuts.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Ixqt/Ixqt-globalkeys/releases/download/2.0.0/Ixqt-globalkeys-2.0.0.tar.xz>
- Download MD5 sum: c919f92964fb62953af831618955f9a3
- Download size: 76 KB
- Estimated disk space required: 12 MB
- Estimated build time: 0.5 SBU

Ixqt-globalkeys Dependencies

Required

[liblxqt-2.0.0](#)

Installation of Ixqt-globalkeys

Install Ixqt-globalkeys by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: lxqt-config-globalkeyshortcuts and lxqt-globalkeysd

Installed Libraries: liblxqt-globalkeys.so and liblxqt-globalkeys-ui.so

Installed Directories: /usr/include/lxqt-globalkeys{-ui} and /usr/share/cmake/lxqt-globalkeys{-ui}

Short Descriptions

lxqt-config-globalkeyshortcuts	is an editor for keyboard shortcuts
lxqt-globalkeysd	is the global keyboard shortcut daemon
lxqt-globalkeys.so	contains the lxqt-globalkeys API functions
liblxqt-globalkeys-ui.so	contains the liblxqt-globalkeys-ui API functions

libdbusmenu-lxqt-0.1.0

Introduction to libdbusmenu-lxqt

The libdbusmenu-lxqt package provides a Qt implementation of the DBusMenu protocol.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/libdbusmenu-lxqt/releases/download/0.1.0/libdbusmenu-lxqt-0.1.0.tar.xz>
- Download MD5 sum: 90b4882c9f84d7dd45129773140942bd
- Download size: 40 KB
- Estimated disk space required: 8.1 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

libdbusmenu-lxqt Dependencies

Required

[CMake-3.30.2](#)

Optional

[qjson](#)

Installation of libdbusmenu-lxqt

Install libdbusmenu-lxqt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -W no-dev .. &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libdbusmenu-lxqt.so

Installed Directory: /usr/lib/cmake/dbusmenu-lxqt

lxqt-qtplugin-2.0.0

Introduction to lxqt-qtplugin

The lxqt-qtplugin package provides an LXQt Qt platform integration plugin.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-qtplugin/releases/download/2.0.0/lxqt-qtplugin-2.0.0.tar.xz>
- Download MD5 sum: 4385d081c09816bdb0ce3d6cd2b31225
- Download size: 24 KB
- Estimated disk space required: 2.4 MB
- Estimated build time: 0.1 SBU

lxqt-qtplugin Dependencies

Required

[liblxqt-2.0.0](#) and [libdbusmenu-lxqt-0.1.0](#)

Installation of lxqt-qtplugin

Install lxqt-qtplugin by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Library: libqtlxqt.so (in \$QT6DIR/plugins/platformthemes)

Installed Directory: \$QT6DIR/plugins/platformthemes

Short Descriptions

`libqtlxqt.so` contains the LXQt QT plugin loader

lxqt-panel-2.0.1

Introduction to lxqt-panel

The lxqt-panel package contains a lightweight X11 desktop panel.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-panel/releases/download/2.0.1/lxqt-panel-2.0.1.tar.xz>
- Download MD5 sum: 8eb8a2ba5293a9c5d7d85f5c31567780

- Download size: 648 KB
- Estimated disk space required: 59 MB
- Estimated build time: 0.5 SBU (Using parallelism=4)

Ixqt-panel Dependencies

Required

[layer-shell-qt-6.1.4](#) for Ixqt, [libdbusmenu-ixqt-0.1.0](#), [liblxqt-2.0.0](#), [lxqt-globalkeys-2.0.0](#), [lxqt-menu-data-2.0.0](#), [solid-6.5.0](#) for Ixqt, and [menu-cache-1.1.0](#)

Recommended

[PulseAudio-17.0](#), [libstatgrab-0.92.1](#) (network monitor and CPU load plugins), [libsystat-1.0.0](#) (CPU and network monitor plugins), [libxkbcommon-1.7.0](#) (keyboard indicator plugin), and [lm-sensors-3-6-0](#) (sensors plugin)

Installation of Ixqt-panel

First, fix a problem identified upstream:

```
sed -e '/lxqt_app_translation_loader/a set(CMAKE_EXECUTABLE_ENABLE_EXPORTS TRUE)' \
-i panel/CMakeLists.txt
```

Install Ixqt-panel by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

- Installed Program:** Ixqt-panel
Installed Libraries: several plugins under /usr/lib/Ixqt-panel
Installed Directories: /usr/lib/Ixqt-panel and /usr/share/Ixqt(panel)

Short Descriptions

`lxqt-panel` is a lightweight Qt6 based panel for the LXQt Desktop

pcmanfm-qt-2.0.0

Introduction to pcmanfm-qt

The pcmanfm-qt is a file manager and desktop icon manager (a Qt port of pcmanfm and libfm).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/Ixqt/pcmanfm-qt/releases/download/2.0.0/pcmanfm-qt-2.0.0.tar.xz>
- Download MD5 sum: 163f512c5ad0994394fd316a0caaef92
- Download size: 388 KB
- Estimated disk space required: 18 MB
- Estimated build time: 0.4 SBU

pcmanfm-qt Dependencies

Required

[layer-shell-qt-6.1.4](#) for [lxqt](#), [liblxqt-2.0.0](#), [libfm-qt-2.0.2](#), and [lxqt-menu-data-2.0.0](#)

Recommended

[Gvfs-1.54.2](#) (runtime) and [oxygen-icons-6.0.0](#) or another icon theme (depending on your choice, some icons will be missing in several places)

Installation of pcmanfm-qt

If libLayerShellQtInterface.so is installed in /opt, help this package find it. As the `root` user:

```
if [ -e /opt/kf6/lib/libLayerShellQtInterface.so -a \
    ! -e /usr/lib/libLayerShellQtInterface.so ]; then

    ln -sv /opt/kf6/lib/libLayerShellQtInterface.so \
        /usr/lib/libLayerShellQtInterface.so
fi
```

Install pcmanfm-qt by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

To make pcmanfm-qt easier to find in menus, adjust the .desktop file. Again, as the `root` user:

```
sed -e '/Categories=/s/=System;FileTools;/' \
      -e '/Name=/s/=.*=File Manager PCManFM-Qt/' \
      -i /usr/share/applications/pcmanfm-qt.desktop
```

Contents

Installed Program: pcmanfm-qt

Installed Library: None

Installed Directories: /usr/share/pcmanfm-qt

Short Descriptions

`pcmanfm-qt` is a file manager and desktop icon manager

lxqt-powermanagement-2.0.0

Introduction to lxqt-powermanagement

The lxqt-powermanagement package provides the power management module for LXQt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-powermanagement/releases/download/2.0.0/lxqt-powermanagement-2.0.0.tar.xz>
- Download MD5 sum: fef1b1f47ff94635ed478e36e4137337

- Download size: 96 KB
- Estimated disk space required: 8.6 MB
- Estimated build time: 0.3 SBU

Ixqt-powermanagement Dependencies

Required

[liblxqt-2.0.0](#), [kidletime-6.5.0 for lxqt](#), [solid-6.5.0 for lxqt](#), and [UPower-1.90.4](#)

Installation of Ixqt-powermanagement

Install Ixqt-powermanagement by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: Ixqt-config-powermanagement and Ixqt-powermanagement

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>lxqt-config-powermanagement</code>	is the <code>lxqt-powermanagement</code> GUI configuration tool
<code>lxqt-powermanagement</code>	is the LXQt power management module

Ixqt-runner-2.0.0

Introduction to Ixqt-runner

The Ixqt-runner package provides a tool used to launch programs quickly by typing their names.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-runner/releases/download/2.0.0/lxqt-runner-2.0.0.tar.xz>
- Download MD5 sum: 923c96c9253fa4b9eb70e1bb0aabfb26
- Download size: 224 KB
- Estimated disk space required: 6.1 MB
- Estimated build time: 0.1 SBU

Ixqt-runner Dependencies

Required

[Ixqt-globalkeys-2.0.0](#), [kwindowsystem-6.5.0 for lxqt](#), and [muparser-2.3.4](#)

Installation of Ixqt-runner

Install lxqt-runner by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Using lxqt-runner

To use lxqt-runner, simultaneously press the Alt-F2 keys. A dialog window appears in the top center of the screen. When you start typing a command, a list of possible matches appears and changes as you keep typing.

Contents

Installed Program: lxqt-runner

Installed Libraries: None

Installed Directories: None

Short Descriptions

`lxqt-runner` launches a graphical dialog box for quickly running a program

Ixqt-themes-2.0.0

Introduction to Ixqt-themes

The Ixqt-themes package provides a number of graphics files and themes for the LXQt desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/Ixqt-themes/releases/download/2.0.0/Ixqt-themes-2.0.0.tar.xz>
- Download MD5 sum: 6e9c345037b4718ff500dfbbac0ce435
- Download size: 25 MB
- Estimated disk space required: 55 MB
- Estimated build time: less than 0.1 SBU

Ixqt-themes Dependencies

Required

[Ixqt-build-tools-2.0.0](#)

Installation of Ixqt-themes

Install Ixqt-themes by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: None

Installed Libraries: None

Installed Directories: /usr/share/lxqt/{graphics,themes,palettes,wallpapers}

lxqt-admin-2.0.0

Introduction to lxqt-admin

The lxqt-admin package provides two GUI tools to adjust settings of the operating system LXQt is running on.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-admin/releases/download/2.0.0/lxqt-admin-2.0.0.tar.xz>
- Download MD5 sum: afc939f22db9e40d45508191a8c7ec04
- Download size: 152 KB
- Estimated disk space required: 4.5 MB
- Estimated build time: 0.2 SBU

lxqt-admin Dependencies

Required

[liblxqt-2.0.0](#) and [Polkit-125](#)

Installation of lxqt-admin

Install lxqt-admin by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: lxqt-admin-time, lxqt-admin-user, and lxqt-admin-user-helper

Installed Libraries: None

Installed Directory: None

Short Descriptions

lxqt-admin-time	is a GUI to adjust the current time and date
lxqt-admin-user	is a GUI to manage users and groups

lxqt-about-2.0.0

Introduction to lxqt-about

The lxqt-about package provides the standalone LXQt “About” dialog.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-about/releases/download/2.0.0/lxqt-about-2.0.0.tar.xz>
- Download MD5 sum: ec17b9089bbf2dc5dfdfbcca196543f6
- Download size: 48 KB
- Estimated disk space required: 3.4 MB
- Estimated build time: less than 0.1 SBU

lxqt-about Dependencies

Required

[liblxqt-2.0.0](#)

Installation of lxqt-about

Install lxqt-about by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: lxqt-about

Installed Libraries: None

Installed Directory: None

Short Descriptions

`lxqt-about` is the LXQt “About” dialog

lxqt-config-2.0.0

Introduction to lxqt-config

The lxqt-config package provides the LXQt Configuration Center.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-config/releases/download/2.0.0/lxqt-config-2.0.0.tar.xz>
- Download MD5 sum: 3c5064b356b4d457258252bd521f3372
- Download size: 372 KB
- Estimated disk space required: 27 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

lxqt-config Dependencies

Required

[liblxqt-2.0.0](#), [kconfig-6.5.0](#) for `lxqt`, [libkscreen-6.1.4](#) for `lxqt`, and [xorg-libinput-1.4.0](#)

Installation of lxqt-config

Install `lxqt-config` by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `lxqt-config`, `lxqt-config-appearance`, `lxqt-config-brightness`, `lxqt-config-file-associations`, `lxqt-config-input`, `lxqt-config-locale`, and `lxqt-config-monitor`

Installed Library: `liblxqt-config-cursor.so`

Installed Directories: `/usr/lib/lxqt-config`

Short Descriptions

<code>lxqt-config</code>	is the LXQt Configuration Center
<code>lxqt-config-appearance</code>	is used to change system themes, icon themes, and fonts used by applications in an LXQt session
<code>lxqt-config-brightness</code>	configures the brightness setting of a monitor
<code>lxqt-config-file-associations</code>	is used to associate filetypes with programs
<code>lxqt-config-input</code>	is used to configure input devices, setting options such as the mouse speed, the cursor, the keyboard speed, and the keyboard layout
<code>lxqt-config-locale</code>	configures the locale used in LXQt sessions
<code>lxqt-config-monitor</code>	is a graphical frontend to <code>xrandr</code> that configures monitors on a system
<code>liblxqt-config-cursor.so</code>	contains the <code>liblxqt-config-cursor</code> API functions

lxqt-policykit-2.0.0

Introduction to lxqt-policykit

The `lxqt-policykit` package is the LXQt PolicyKit agent.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-policykit/releases/download/2.0.0/lxqt-policykit-2.0.0.tar.xz>
- Download MD5 sum: 4c3b3533a966798fb46ff8da3f8ffa07
- Download size: 40 KB
- Estimated disk space required: 2.6 MB
- Estimated build time: less than 0.1 SBU

lxqt-policykit Dependencies

Required

[liblxqt-2.0.0](#) and [polkit-qt-0.200.0](#)

Installation of lxqt-policykit

Install lxqt-policykit by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: lxqt-policykit-agent

Installed Libraries: None

Installed Directory: /usr/share/lxqt/translations/lxqt-policykit-agent

Short Descriptions

`lxqt-policykit-agent` is the LXQt PolicyKit agent

lxqt-sudo-2.0.0

Introduction to lxqt-sudo

The lxqt-sudo package is a graphical front end to sudo and su respectively.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-sudo/releases/download/2.0.0/lxqt-sudo-2.0.0.tar.xz>
- Download MD5 sum: e6e2079901270e6f9b527dc5d44e098e
- Download size: 40 KB
- Estimated disk space required: 2.4 MB
- Estimated build time: less than 0.1 SBU

lxqt-sudo Dependencies

Required

[liblxqt-2.0.0](#)

Installation of lxqt-sudo

Install lxqt-sudo by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `lxdoas` (symlinkk to `lxqt-sudo`), `lxqt-sudo`, `lxsu` (symlinkk to `lxqt-sudo`), and `lxsudo` (symlinkk to `lxqt-sudo`)

Installed Libraries: None

Installed Directories: /usr/share/lxqt/translations/lxqt-sudo

Short Descriptions

<code>lxdoas</code>	is a graphical frontend for <code>doas</code>
<code>lxqt-sudo</code>	is a graphical frontend for <code>sudo</code>
<code>lxsudo</code>	is a graphical frontend for <code>sudo</code>
<code>lxsu</code>	is a graphical frontend for <code>su</code>

Ixqt-openssh-askpass-2.0.1

Introduction to Ixqt-openssh-askpass

The `lxqt-openssh-askpass` package is a GUI to query credentials on behalf of other programs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-openssh-askpass/releases/download/2.0.1/lxqt-openssh-askpass-2.0.1.tar.xz>
- Download MD5 sum: 7ce91af596704385f206db07f726c26e
- Download size: 20 KB
- Estimated disk space required: 1.7 MB
- Estimated build time: less than 0.1 SBU

Ixqt-openssh-askpass Dependencies

Required

[liblxqt-2.0.0](#)

Installation of Ixqt-openssh-askpass

Install `lxqt-openssh-askpass` by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `lxqt-openssh-askpass`

Installed Libraries: None

Installed Directories: /usr/share/lxqt/translations/lxqt.openssh-askpass

Short Descriptions

lxqt-config-openssh-askpass queries a user for credentials on behalf of another program

lxqt-session-2.0.0

Introduction to lxqt-session

The lxqt-session package contains the default session manager for LXQt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-session/releases/download/2.0.0/lxqt-session-2.0.0.tar.xz>
- Download MD5 sum: 7e3e0553e7562ef5c25072c8dc671f78
- Download size: 192 KB
- Estimated disk space required: 12 MB
- Estimated build time: 0.4 SBU

lxqt-session Dependencies

Required

[liblxqt-2.0.0](#), [qtxdg-tools-4.0.0](#), and [Xdg-user-dirs-0.18](#)

Installation of lxqt-session

Install lxqt-session by running the following commands:

```
sed -e '/TryExec/s|=|= /usr/bin/|' \
    -i xsession/lxqt.desktop.in &&

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed -e '/TryExec/s|=|= /usr/bin/|' -i xsessions/lxqt.desktop.in`: If you are using a display manager, the full path of the "TryExec" directive has to be given so that the LXQt desktop will appear in the sessions list.

Configuring lxqt-session

Window Managers other than [openbox-3.6.1](#) may be used, e.g. [Xfwm4-4.18.0](#). Please note that [IceWM-3.6.0](#) is not a good substitute. Fluxbox does work, although in this context (with [lxqt-config-2.0.0](#)), [openbox-3.6.1](#) is better. The configuration file `/usr/share/lxqt/windowmanagers.conf` comes with many examples of Window Managers and the ones which are installed will appear in a drop down list of `lxqt-config-session`. For the ones not included in `/usr/share/lxqt/windowmanagers.conf`, you can use `lxqt-config-session`'s "search" button, e.g. for [Fluxbox-1.3.7](#), navigating through the file system until you can choose `fluxbox`.

Contents

Installed Programs: lxqt-config-session, lxqt-leave, lxqt-session, and startlxqt

Installed Libraries: None

Installed Directories: /usr/share/lxqt/translations/lxqt-{leave,config-session,session}

Short Descriptions

lxqt-config-session	is a GUI configuration tool for lxqt-session
lxqt-leave	is a graphical dialog to terminate the session
lxqt-session	is a lightweight X session manager
startlxqt	is used to start the desktop session for LXQt

xdg-desktop-portal-lxqt-1.0.2

Introduction to xdg-desktop-portal-lxqt

xdg-desktop-portal-lxqt is a backend for xdg-desktop-portal, that is using the Qt library.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/xdg-desktop-portal-lxqt/releases/download/1.0.2/xdg-desktop-portal-lxqt-1.0.2.tar.xz>
- Download MD5 sum: 005b8fe696067d1177222a8dfd1866c9
- Download size: 20 KB
- Estimated disk space required: 2.1 MB
- Estimated build time: less than 0.1 SBU

xdg-desktop-portal-lxqt Dependencies

Required

[libfm-qt-2.0.2](#), [kwindowsystem-6.5.0 for lxqt](#), and [xdg-desktop-portal-1.18.2](#) (at runtime)

Installation of xdg-desktop-portal-lxqt

Install xdg-desktop-portal-lxqt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: xdg-desktop-portal-lxqt (in /usr/libexec)

Installed Library: None

Installed Directory: /usr/share/xdg-desktop-portal (if no other xdg-desktop-portal backend is installed)

obconf-qt-0.16.4

Introduction to obconf-qt

The obconf-qt package is a Qt-based configuration tool for Openbox.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/obconf-qt/releases/download/0.16.4/obconf-qt-0.16.4.tar.xz>
- Download MD5 sum: e89509d190e34fef75875856edcb4855
- Download size: 116 KB
- Estimated disk space required: 4.6 MB
- Estimated build time: 0.2 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/obconf-qt-0.16.4-qt6-1.patch>

obconf-qt Dependencies

Required

[hicolor-icon-theme-0.18](#), [lxqt-build-tools-2.0.0](#), [openbox-3.6.1](#), and [Qt-6.7.2](#)

Installation of obconf-qt

First, make the program compatible with [Qt-6.7.2](#).

```
patch -Np1 -i ../obconf-qt-0.16.4-qt6-1.patch
```

Install obconf-qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: obconf-qt

Installed Libraries: None

Installed Directory: None

Short Descriptions

`obconf-qt` is a Qt-based configuration tool for Openbox

LXQt Desktop Final Instructions

Post-Install Instructions

Please follow these instructions before starting LXQt for the first time.

Dependencies to Start LXQt

Required

[openbox-3.6.1](#), or another window manager, such as [Xfwm4-4.18.0](#), or kwin from [Plasma-6.1.4](#). Note that [IceWM-3.6.0](#) is not suitable for LXQt.

Recommended

[breeze-icons-6.5.0](#) and [desktop-file-utils-0.27](#)

Optional

[lightdm-1.32.0](#) or another Display Manager, e.g. [sddm-0.21.0](#), or [XScreenSaver-6.09](#)

Final Configuration

LXQt Final Database Updates

The desktop databases need to be created or updated at this point. Run the following commands as the `root` user:

```
ldconfig &&
update-mime-database /usr/share/mime &&
xdg-icon-resource forceupdate &&
update-desktop-database -q
```

Starting LXQt

You can start LXQt from a [TTY](#), using [xinit-1.4.2](#), or [by](#) using a [graphical display manager](#) such as [lightdm-1.32.0](#).

To start LXQt using [xinit-1.4.2](#), run the following commands:

```
cat > ~/.xinitrc << "EOF"
exec startlxqt
EOF

startx
```

The X session starts on the first unused virtual terminal, normally vt1. You can switch to another `vtn` by simultaneously pressing the keys `Ctrl-Alt-Fn` ($n=2, 3, \dots$). To switch back to the X session, normally started at vt1, use `Ctrl-Alt-F1`. The vt where the command `startx` was executed will display many messages, including X starting messages, applications automatically started with the session, and eventually, some warning and error messages, but these are hidden by the graphical interface. You may prefer to redirect those messages to a log file, which can be used for debugging purposes. This can be done starting X with:

```
startx &> ~/.x-session-errors
```

Initial Configuration

When LXQt first starts, it will ask you for the window manager to use. To start, the BLFS editors recommend using openbox. At this point both the background and the panel will be black. Right clicking on the background will bring up a menu and selecting "Desktop Preferences" will allow you to change the background color or set a background image.

The panel will be at the bottom of the screen. Right clicking on the panel will bring up a menu that will allow you to customize the panel including adding widgets and setting the background color. The BLFS editors recommend installing, at a minimum, the Application Manager and Task Manager widgets.

After LXQt has started for the first time, the BLFS editors recommend going through the settings presented in the LXQt Configuration Center, which can be found under LXQt Settings in the Preferences menu of the application launcher.

Note

The user configuration files will be created in the directory `$HOME/.config/lxqt/`. To get widget icons to display properly, the `lxqt.conf` file may need to be manually edited to include the line "icon_theme=oxygen".

Chapter 38. LXQt Applications

This is a small collection of optional applications that add extra capabilities to the LXQt desktop.

Iximage-qt-2.0.1

Introduction to Iximage-qt

The Iximage-qt package contains a lightweight image viewer and screenshot program.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/Iximage-qt/releases/download/2.0.1/Iximage-qt-2.0.1.tar.xz>
- Download MD5 sum: 50e5c4167fb5b6cde427199f07f1a30b
- Download size: 832 KB
- Estimated disk space required: 8.3 MB
- Estimated build time: 0.3 SBU

Iximage-qt Dependencies

Required

[libfm-qt-2.0.2](#)

Installation of Iximage-qt

Install Iximage-qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_PREFIX=/usr \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have [xdg-utils-1.2.1](#) installed, as the `root` user, run `xdg-icon-resource forceupdate --theme hicolor`.

Contents

Installed Program: Iximage-qt

Installed Libraries: None

Installed Directory: /usr/share/Iximage-qt

Short Descriptions

`lximage-qt` is a lightweight image viewer and screenshot program. There is a menu entry in Utilities for the screenshot program. From the command line, you can obtain a full screen or window shot by running `lximage-qt --screenshot`

Ixqt-archiver-1.0.0

Introduction to Ixqt-archiver

The Ixqt-archiver package is a simple and lightweight desktop-agnostic Qt file archiver.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-archiver/releases/download/1.0.0/lxqt-archiver-1.0.0.tar.xz>
- Download MD5 sum: 3145a209ebe5e9e97468a102e1b8396c
- Download size: 184 KB
- Estimated disk space required: 11 MB
- Estimated build time: 0.2 SBU

lxqt-archiver Dependencies

Required

[libfm-qt-2.0.2](#), [liblxqt-2.0.0](#), and [JSON-GLib-1.8.0](#)

Installation of lxqt-archiver

Install lxqt-archiver by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_BUILD_TYPE=Release \
      -D CMAKE_INSTALL_PREFIX=/usr \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: lxqt-archiver

Installed Libraries: None

Installed Directories: /usr/share/lxqt-archiver

Short Descriptions

`lxqt-archiver` is a Qt-based file archiver

lxqt-notificationd-2.0.1

Introduction to lxqt-notificationd

The lxqt-notificationd package is the LXQt notification daemon.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/lxqt-notificationd/releases/download/2.0.1/lxqt-notificationd-2.0.1.tar.xz>
- Download MD5 sum: 68c41fc8bf83d9620f42cb159261ecbc
- Download size: 64 KB
- Estimated disk space required: 5.1 MB
- Estimated build time: 0.2 SBU

lxqt-notificationd Dependencies

Required

[liblxqt-2.0.0](#)

Installation of lxqt-notificationd

Install lxqt-notificationd by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_PREFIX=/usr \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: lxqt-config-notificationd and lxqt-notificationd

Installed Libraries: None

Installed Directories: /usr/share/lxqt/translations/lxqt-config-notificationd and /usr/share/lxqt/translations/lxqt-notificationd

Short Descriptions

lxqt-config-notificationd	is the <code>lxqt-notificationd</code> GUI configuration tool
lxqt-notificationd	is the LXQt notification daemon

pavucontrol-qt-2.0.0

Introduction to pavucontrol-qt

pavucontrol-qt is the Qt port of the volume control utility pavucontrol, which is used to adjust audio levels in Pulseaudio. It is desktop environment agnostic.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/pavucontrol-qt/releases/download/2.0.0/pavucontrol-qt-2.0.0.tar.xz>
- Download MD5 sum: d530390b798e299aa42e9b81cda033b7
- Download size: 112 KB
- Estimated disk space required: 7.4 MB
- Estimated build time: 0.2 SBU

pavucontrol-qt Dependencies

Required

[liblxqt-2.0.0](#) and [PulseAudio-17.0](#) (built with [GLib-2.80.4](#) support)

Installation of pavucontrol-qt

Install pavucontrol-qt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_PREFIX=/usr \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: pavucontrol-qt

Installed Libraries: None

Installed Directory: /usr/share/pavucontrol-qt

Short Descriptions

`pavucontrol-qt` is a Qt based GUI configuration tool for sound settings using pulseaudio

qps-2.9.0

Introduction to qps

The qps package contains a Qt process manager.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/qps/releases/download/2.9.0/qps-2.9.0.tar.xz>
- Download MD5 sum: 3f5465fa6ed6b02d75c8bf2553f54cc0
- Download size: 448 KB
- Estimated disk space required: 11 MB
- Estimated build time: 0.4 SBU

qps Dependencies

Required

[liblxqt-2.0.0](#)

Installation of qps

Install qps by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_BUILD_TYPE=Release \
-D CMAKE_INSTALL_PREFIX=/usr \
..
&&

make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have [xdg-utils-1.2.1](#) installed, as the `root` user, run `xdg-icon-resource forceupdate --theme hicolor`.

Contents

Installed Program: qps

Installed Libraries: None

Installed Directory: /usr/share/qps

Short Descriptions

qtermwidget-2.0.1

Introduction to qtermwidget

As the name suggests, the qtermwidget package is a terminal widget for Qt.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/qtermwidget/releases/download/2.0.1/qtermwidget-2.0.1.tar.xz>
- Download MD5 sum: fe548f62dcc65064a84a54cbd133a0be
- Download size: 188 KB
- Estimated disk space required: 6.7 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

qtermwidget Dependencies

Required

[Qt-6.7.2](#)

Installation of qtermwidget

Install qtermwidget by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: None

Installed Libraries: libqtermwidget5.so

Installed Directories: /usr/include/qtermwidget5, /usr/lib/cmake/qtermwidget5, and /usr/share/qtermwidget5

Short Descriptions

`libqtermwidget5.so` provides a terminal widget for Qt5

qterminal-2.0.1

Introduction to qterminal

The qterminal package contains a Qt widget based terminal emulator for Qt with support for multiple tabs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/qterminal/releases/download/2.0.1/qterminal-2.0.1.tar.xz>

- Download MD5 sum: 56bf4e1dbd950c457c8b9af4a0644bb8
- Download size: 228 KB
- Estimated disk space required: 10 MB
- Estimated build time: 0.3 SBU (Using parallelism=4)

qterminal Dependencies

Required

[qtermwidget-2.0.1](#)

Installation of qterminal

Install qterminal by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
..
&&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: qterminal

Installed Libraries: None

Installed Directories: /usr/share/qterminal

Short Descriptions

`qterminal` is a Qt based lightweight terminal emulator

screengrab-2.8.0

Introduction to screengrab

The screengrab package is a cross-platform tool for making screenshots quickly.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/lxqt/screengrab/releases/download/2.8.0/screengrab-2.8.0.tar.xz>
- Download MD5 sum: bec5465f053bfb964c7a6e57e20e421f
- Download size: 340 KB
- Estimated disk space required: 9.1 MB
- Estimated build time: 0.2 SBU

screengrab Dependencies

Required

[kwindowsystem-6.5.0](#) for lxqt and [libqtxdg-4.0.0](#)

Installation of screengrab

Install screengrab by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_BUILD_TYPE=Release \
      -D CMAKE_INSTALL_PREFIX=/usr \
      ..
      &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have [xdg-utils-1.2.1](#) installed, as the `root` user, run `xdg-icon-resource forceupdate --theme hicolor`.

Contents

Installed Program: screengrab

Installed Libraries: None

Installed Directory: /usr/share/screengrab and /usr/share/doc/screengrab

Short Descriptions

`screengrab` is a cross platform tool for creating screenshots quickly

Part XI. X Software

Chapter 39. Office Programs

This chapter is a collection of programs that are useful for viewing or editing office documents. Some specialise in doing one thing (such as word processing or manipulating a spreadsheet). LibreOffice is a suite of programs that can manipulate many different formats including powerpoint presentations.

AbiWord-3.0.5

Introduction to AbiWord

AbiWord is a word processor which is useful for writing reports, letters and other formatted documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/abiword/abiword-3.0.5.tar.gz>
- Download MD5 sum: a8f218b711450e4ccae43a0522e0e806
- Download size: 11 MB
- Estimated disk space required: 690 MB, with docs (76 MB installed)
- Estimated build time: 2.4 SBU (Using parallelism=4, with docs)

Additional Downloads

- AbiWord Docs: <https://anduin.linuxfromscratch.org/BLFS/abiword/abiword-docs-3.0.2.tar.gz>
- AbiWord Docs MD5 sum: c92c0e1e2081db20757231a53d80c338
- AbiWord Docs size: 1.5 MB

AbiWord Dependencies

Required

[Boost-1.86.0](#), [cURL-8.9.1](#), [FriBidi-1.0.15](#), [GOffice-0.10.57](#), and [wv-1.2.9](#)

Recommended

[enchant-2.8.2](#)

Optional

[dbus-glib-0.112](#), [evolution-data-server-3.52.4](#), [GLib-2.80.4](#) (with GObject Introspection), [libgcrypt-1.11.0](#), [libical-3.0.18](#), [libsoup-2.74.3](#), [Redland-1.0.17](#), [Valgrind-3.23.0](#). [Aiksaurus](#), [GtkMathView](#), [libchamplain](#), [libwmf](#), [libwpd](#), [libwpg](#), [libwps](#), [Link Grammar Parser](#), [Loudmouth](#), [OTS](#), [Psiconv](#), and [telepathy-glib](#)

Note

To enable many of the optional dependencies, review the information from `./configure --help` for the necessary switches you must pass to the `configure` script.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/abiword>

Installation of AbiWord

First, fix a build failure when using libxml-2.12.x:

```
sed -e '/libxml/a #include <libxml/xmlmemory.h>' \
      -i src/af/util/xp/ut_stringbuf.cpp
```

Install AbiWord by running the following commands:

```
sed -e "s/free_suggestions/free_string_list/" \
      -e "s/_to_personal//"
      -e "s/in_session/added/"
      -i src/af/xap/xp/enchant_checker.cpp    &&
./configure --prefix=/usr &&
make
```

This program does not come with a functional test suite.

Now, as the `root` user:

```
make install
```

Note

In order to install the local help files, a graphics session is recommended.

If you wish to install the local help files, untar and build them first:

Note

The current version of the docs expands to abiword-docs-3.0.1 and not the expected abiword-docs-3.0.2.

```
tar -xf ../abiword-docs-3.0.2.tar.gz &&
cd abiword-docs-3.0.1                &&
./configure --prefix=/usr              &&
make
```

and then, as the `root` user:

```
make install
```

Command Explanations

`sed ...`: This command corrects the source for changes in the dependency [enchant-2.8.2](#).

--enable-plugins="collab openxml goffice grammar": Build some or all plugins. The openxml plugin enables Abiword to open some .docx files. The grammar plugin requires [Link Grammar Parser](#)

Configuring AbiWord

Config File

~/.AbiSuite/templates/normal.awt

Configuration Information

Choose the right template for your language and locale from the list produced by the following command:

```
ls /usr/share/abiword-3.0/templates
```

Create the folder `~/.AbiSuite/templates` then copy the `normal.awt` you want into it:

```
install -v -m750 -d ~/.AbiSuite/templates &&
install -v -m640      /usr/share/abiword-3.0/templates/normal.awt-<lang> \
                     ~/.AbiSuite/templates/normal.awt
```

Change `<lang>` by the above command to fit the name of the file you want.

If you are using multiple languages, you may need to edit the template to use a font with greater coverage (e.g. one of the [Dejavu fonts](#), because Abiword does not use fontconfig and can only display glyphs that are provided in the chosen font).

If you have [desktop-file-utils-0.27](#) installed, you should run the `update-desktop-database` command to update the mimeinfo cache and allow the Help system to work.

If you have [xdg-utils-1.2.1](#) installed, you should run the `xdg-icon-resource forceupdate --theme hicolor` command, for the installed icon to be displayed in the menu item.

Contents

Installed Program: abiword

Installed Library: libabiword-3.0.so

Installed Directories: /usr/include/abiword-3.0, /usr/lib/abiword-3.0, and /usr/share/abiword-3.0

Short Descriptions

abiword	is the word processor, a wrapper for the functions in libabiword-3.0 - it can also be used on the command line, see <code>man 1 abiword</code>
libabiword-3.0.so	provides functions to access MS Word documents

Gnumeric-1.12.57

Introduction to Gnumeric

The Gnumeric package contains a spreadsheet program which is useful for mathematical analysis.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/gnumeric/1.12/gnumeric-1.12.57.tar.xz>
- Download MD5 sum: 237ce4f6c43661892e802bf81dbff1b2
- Download size: 17 MB
- Estimated disk space required: 244 MB (add 4 MB for tests)
- Estimated build time: 0.8 SBU (add 1.7 SBU for tests; both using parallelism=4)

Gnumeric Dependencies

Required

[GOffice-0.10.57](#) and [itstool-2.0.7](#)

Recommended Runtime

an icon theme such as [adwaita-icon-theme-46.2](#), [oxygen-icons-6.0.0](#), or [gnome-icon-theme-3.12.0](#), [a graphical environment](#), and [Yelp-42.2](#) for the built-in help functionality

Optional

[DConf-0.40.0](#) (to prevent some test failures), [GLib-2.80.4](#) (with GObject Introspection), [PyGObject-3.48.2](#), [Valgrind-3.23.0](#) (for some tests), [dblatex](#) (for PDF docs), [libgda](#), [Mono](#), [pxlib](#), and [Psiconv](#)

Installation of Gnumeric

Install Gnumeric by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package must be installed before the test suite is run.

Now, as the `root` user:

```
make install
```

To run the tests, issue `make check`. Tests should be run from an active X session, and [DConf-0.40.0](#) should be installed to prevent some failures. Two tests are known to fail. Additional failures may occur if valgrind is installed.

Command Explanations

`--enable-pdfdocs`: Use this switch if you have installed dblatex and wish to create PDF docs.

Contents

Installed Programs: gnumeric (symlink), gnumeric-1.12.57, ssconvert, ssdiff, ssgrep, and ssindex

Installed Libraries: libspreadsheet.so

Installed Directories: /usr/include/libspreadsheet-1.12, /usr/lib/gnumeric, /usr/lib/goffice/0.10/plugins/gnumeric, /usr/share/gnumeric, and /usr/share/help/{C,cs,de,es}/gnumeric

Short Descriptions

<code>gnumeric</code>	is a symlink to <code>gnumeric-1.12.57</code>
<code>gnumeric-1.12.57</code>	is GNOME's spreadsheet application
<code>ssconvert</code>	is a command line utility to convert spreadsheet files between various spreadsheet file formats
<code>ssdiff</code>	is a command line utility to compare two spreadsheets
<code>ssgrep</code>	is a command line utility to search spreadsheets for strings
<code>ssindex</code>	is a command line utility to generate index data for spreadsheet files
<code>libspreadsheet.so</code>	contains the gnumeric API functions

LibreOffice-24.8.0

Introduction to LibreOffice

LibreOffice is a full-featured office suite. It is largely compatible with Microsoft Office and is descended from OpenOffice.org.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Core Download: <https://download.documentfoundation.org/libreoffice/src/24.8.0/libreoffice-24.8.0.3.tar.xz>
- Core Download MD5 sum: 230fb890c1fc32b8e7186a96d4e9c12a
- Core Download size: 267 MB

- Estimated disk space required: 11 GB including the additional files if downloaded during the build (775 MB installed). Installing the whole set of languages requires more than 25 GB (around 3 GB installed)
- Estimated build time: 32 SBU with parallelism=8 (will vary due to download time)

Additional Downloads

Note

Around 80 smallish tarballs and zip files will be downloaded during the build. The download time will vary with the local internet speed and server load. The download time is about 3 minutes on a fiber connection and probably not much more than 10 minutes on an ADSL connection. If you have not installed one or more of the following dependencies, they will also be downloaded during the build.

- Dictionaries: <https://download.documentfoundation.org/libreoffice/src/24.8.0/libreoffice-dictionaries-24.8.0.3.tar.xz>
- Dictionaries MD5 sum: aabc278e0393220c39cb72a98e56abec
- Dictionaries size: 56 MB
- Help Files: <https://download.documentfoundation.org/libreoffice/src/24.8.0/libreoffice-help-24.8.0.3.tar.xz>
- Help Files MD5 sum: f5b9ae5cc8572f01a6ae66704ce790bf
- Help Files size: 160 MB
- Translations: <https://download.documentfoundation.org/libreoffice/src/24.8.0/libreoffice-translations-24.8.0.3.tar.xz>
- Translations MD5 sum: d86da685bce6182f77fc4002ad8d9f8f
- Translations size: 212 MB
- Required patch to fix building with Boost-1.86.0:
https://www.linuxfromscratch.org/patches/blfs/12.2/libreoffice-24.8.0.3-boost_fixes-1.patch

LibreOffice Dependencies

Required

[Archive-Zip-1.68](#), [UnZip-6.0](#), [Wget-1.24.5](#), [Which-2.21](#), and [Zip-3.0](#)

Note

An Internet connection is needed for building this package.

Recommended

Note

Most of these packages are recommended because if they're not installed, the build process will compile and install its own (often older) version, with the side effect of increasing both build and installed disk space along with increasing build time.

[apache-ant-1.10.14](#), [Boost-1.86.0](#), [CLucene-2.3.3.4](#), [Cups-2.4.10](#), [cURL-8.9.1](#), [libepoxy-1.5.10](#), [libjpeg-turbo-3.0.1](#), [LLVM-18.1.7](#) (clang is preferred for rendering with skia), [GLM-1.0.1](#), [GLU-9.0.3](#), [GPGME-1.23.2](#), [Graphite2-1.3.14](#), [gst-plugins-base-1.24.7](#), [GTK+-3.24.43](#), [harfBuzz-9.0.0](#), [ICU-75.1](#), [libatomic_ops-7.8.2](#), [Little CMS-2.16](#), [librsvg-2.58.3](#), [libtiff-4.6.0](#), [libwebp-1.4.0](#), [libxml2-2.13.3](#) and [libxslt-1.1.42](#), [lxml-5.3.0](#), [Mesa-24.1.5](#), [nss-3.103](#), [OpenLDAP-2.6.8](#) (if connecting to an LDAP server), [Poppler-24.08.0](#) (currently broken), [PostgreSQL-16.4](#), [Redland-1.0.17](#), and [unixODBC-2.3.12](#)

Optional

[Abseil-cpp-20240722.0](#), [Avahi-0.8](#), [BlueZ-5.77](#), [DConf-0.40.0](#), [desktop-file-utils-0.27](#), [Doxygen-1.12.0](#) (not relevant if using --disable-odk), [evolution-data-server-3.52.4](#), [GDB-15.1](#), [GnuTLS-3.8.7.1](#), [KDE Frameworks-6.5.0](#), [libpaper-2.2.5](#), [MariaDB-10.11.8](#) or [MySQL](#), [MIT Kerberos V5-1.21.3](#), [NASM-2.16.03](#), [Qt-6.7.2](#), [SANE-1.2.1](#), [Valgrind-3.23.0](#), [VLC-3.0.21](#), [Apache Commons Codec](#), [Apache Commons HttpClient](#), [Apache Commons Lang](#), [Apache Commons Logging](#), [BeanShell](#), [box2d](#), [CoinMP](#), [Cppunit](#), [Firebird](#), [glyphy](#), [Gnome VFS](#), [hamcrest](#), [Hunspell](#), [Hyphen](#), [iwyu](#), [junit4](#), [libabw](#), [libcdr](#), [libcmis](#), [libebook](#), [libepubgen](#), [libetonyek](#), [libexttextcat](#), [libfreehand](#), [liblangtag](#), [libmspub](#), [libmwaw](#), [libnumbertext](#), [libodfgen](#), [libpagemaker](#), [librgcdegen](#), [libqxp](#), [librevenge](#) ([WordPerfect Document importer](#)),

[libstaroffice](#), [libvisio](#), [libwpd](#), [libwpg](#), [libwps](#), [libzmf](#), [lp_solve](#), [mdds](#), [MyThes](#), [odfvalidator](#), [officeotron](#), [OpenCOLLADA](#), [Orcus](#), [SystemTap](#), and [VIGRA](#)

There are many optional dependencies not listed here. They can be found in the download.1st file in the sources directory.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/libreoffice>

Installation of LibreOffice

First, fix build failures introduced by the latest Boost version:

```
patch -Np1 -i ../libreoffice-24.8.0.3-boost_fixes-1.patch
```

Next, fix a bug with linking to zlib:

```
sed -i '/icuuc \\\a zlib\\\' writerperfect/Library_wpfdraw.mk
```

If you have downloaded the dictionaries, help and translations tarballs, create symlinks to them from the source directory so they won't get downloaded again:

```
install -dm755 external/tarballs &&
ln -sv ../../..../libreoffice-dictionaries-24.8.0.3.tar.xz external/tarballs/ &&
ln -sv ../../..../libreoffice-help-24.8.0.3.tar.xz external/tarballs/ &&
ln -sv ../../..../libreoffice-translations-24.8.0.3.tar.xz external/tarballs/
```

The instructions in the package unpack some tarballs into a location it cannot find later. Create some symlinks to help the build system out:

```
ln -sv src/libreoffice-help-24.8.0.3/helpcontent2/ &&
ln -sv src/libreoffice-dictionaries-24.8.0.3/dictionaries/ &&
ln -sv src/libreoffice-translations-24.8.0.3/translations/
```

Note

During the build process, some packages will be downloaded (including the ones listed as recommended and optional dependencies) if they are not present on the system. Because of this, build time may vary from the published time more than usual.

Due to the large size of the package, you may prefer to install it in `/opt`, instead of `/usr`. Depending on your choice, replace `<PREFIX>` by `/usr` or by `/opt/libreoffice-24.8.0.3`:

```
export LO_PREFIX=<PREFIX>
```

Locales "fr" and "en-GB", you will find below, are just examples; you should change them to suit your needs - you might want to read the "Command Explanations", further below, before proceeding.

Note

If you set the ACLOCAL environment variable to support installing Xorg in `/opt`, you will need to unset it for this package.

If you are building on a 32 bit machine, `CFLAGS` is set to `-os`, which breaks the build. Prevent this by issuing:

```
case $(uname -m) in
  i?86) sed -Os/d -i solenv/gbuild/platform/LINUX_INTEL_GCC.mk ;;
esac
```

Prepare LibreOffice for compilation by running the following commands:

```
sed -e "/gzip -f/d" \
      -e "s|.1.gz|.1|g" \
      -i bin/distro-install-desktop-integration &&

sed -e "/distro-install-file-lists/d" -i Makefile.in &&

./autogen.sh --prefix=$LO_PREFIX \
             --sysconfdir=/etc \

```

```
--with-vendor=BLFS          \
--with-lang='fr en-GB'       \
--with-help                  \
--with-myspell-dicts        \
--without-junit              \
--without-system-dicts       \
--disable-dconf              \
--disable-odk                \
--disable-mariadb-sdbc       \
--enable-release-build=yes   \
--enable-python=system        \
--with-jdk-home=/opt/jdk     \
--with-system-boost          \
--with-system-clucene         \
--with-system-curl            \
--with-system-epoxy           \
--with-system-expat           \
--with-system-glm              \
--with-system-gpgmepp         \
--with-system-graphite        \
--with-system-harfbuzz         \
--with-system-jpeg              \
--with-system-lcms2             \
--with-system-libatomic_ops    \
--with-system-libpng            \
--with-system-libxml            \
--with-system-nss              \
--with-system-odbc              \
--with-system-openldap          \
--with-system-openssl           \
--with-system-poppler          \
--with-system-postgresql        \
--with-system-redland           \
--with-system-libtiff            \
--with-system-libwebp            \
--with-system-icu                \
--with-system-zlib
```

Build the package:

```
make build
```

Now, as the `root` user:

```
make distro-pack-install
```

If installed in `/opt/libreoffice-24.8.0.3` some additional steps are necessary. Issue the following commands as the `root` user:

```
if [ "$LO_PREFIX" != "/usr" ]; then

    # This symlink is necessary for the desktop menu entries
    ln -svf $LO_PREFIX/lib/libreoffice/program/soffice /usr/bin/libreoffice &&

    # Set up a generic location independent of version number
    ln -sfv $LO_PREFIX /opt/libreoffice

    # Icons
    mkdir -vp /usr/share/pixmaps
    for i in $LO_PREFIX/share/icons/hicolor/32x32/apps/*; do
        ln -svf $i /usr/share/pixmaps
    done &&

    # Desktop menu entries
    for i in $LO_PREFIX/lib/libreoffice/share/xdg/*; do
        ln -svf $i /usr/share/applications/libreoffice-$(basename $i)
    done &&

    # Man pages
    for i in $LO_PREFIX/share/man/man1/*; do
        ln -svf $i /usr/share/man/man1/
    done

    unset i
fi
```

If you have installed [desktop-file-utils-0.27](#), and you wish to update the MIME database, issue, as the `root` user:

```
update-desktop-database
```

The suite and main modules can be started from the menu. From a terminal, the suite can be started with the `libreoffice` command and the modules with the `libreoffice --<module>` command, respectively, where `<module>` is one of base, calc, draw, impress, math, web, or writer. Modules cannot be started using their shell starting script names (see "Installed Programs", below), if `LO_PREFIX` is other than `/usr`, unless the `$LO_PREFIX/bin` directory is appended to the PATH.

Command Explanations

`sed -e ...`: The first sed prevents compression of the manual pages and the second one prevents the install from failing.

`--with-vendor=BLFS`: This switch sets BLFS as the vendor which is mentioned when you click "About" from the Help menu on the toolbar.

`--with-lang='fr en-GB'`: This switch sets what languages to support. To list several languages, separate them with a space. For all languages, use `--with-lang=ALL`. Note that the translations file is not needed and won't be downloaded if using only en-US as a language.

Note

For a list of the available languages, look in `solenv/inc/langlist.mk`.

`--with-help`: Without this switch, the help files are not built.

`--with-jdk-home=/opt/jdk`: This parameter will silence a warning that the configure script attempted to find `JAVA_HOME` automatically (even though that is passed in the environment). Omit this if you disabled java.

`--with-myspell-dicts`: This switch adds myspell dictionaries to the LibreOffice installation set.

`--with-system-boost`: This switch enables using system boost. From time to time, recent versions of boost break the build of libreoffice. In this case, it is possible to use the bundled copy of boost by removing this flag.

`--disable-dconf`: This switch disables compiling LibreOffice with the GNOME DConf configuration system support.

`--without-junit`: This switch disables the tests for the deprecated HSQLDB driver which is written in Java.

`--without-system-dicts`: This switch disables the use of dictionaries from system paths so that the ones installed by this package are used.

`--disable-odk`: This switch disables installing the office development kit. Remove if you want to develop a LibreOffice based application.

`--disable-mariadb-sdbc`: This switch disables the MariaDB SDBC connector for Libreoffice Base. Leaving it enabled results in a build failure because of a problem linking to [MIT Kerberos V5-1.21.3](#).

`--enable-release-build=yes`: This switch enables a Release Build. LibreOffice can be built as a Release Build or as a Developer Build, because their default installation paths and user profile paths are different. Developer Build displays the words "Dev" and "Beta" in several places (e.g., menu and splash screen).

`--enable-python=system`: This switch tells LibreOffice to use the system installed Python 3 instead of the bundled one.

`--with-system-icu`: This switch tells the build system to use the system installed version of ICU.

`--with-system-*`: These switches prevent LibreOffice from trying to compile its own versions of these dependencies. If you've not installed some of the dependencies, remove the corresponding switches.

`make distro-pack-install`: This does a normal install, but if you add a DESTDIR environment variable it will also install a lot of (text) `gid_Module_*` files in the DESTDIR, to help distributions which want to break the package into parts.

`--with-parallelism=<count>`: This switch tells LibreOffice to use <count> CPU cores to compile in parallel. (Do not include literal angle brackets.) The default is to use all available cores on the system.

`--disable-cups`: Use this switch if you don't need printing support.

`--disable-dbus`: Use this switch if you've not installed D-Bus-1.8.0 or later. It also disables Bluetooth support and font install via PackageKit.

`--disable-firebird-sdbc`: By default, the ability to connect to a firebird database is enabled. Add this switch if you do not need that feature.

--disable-gstreamer-1-0: Use this switch if you have not installed [gst-plugins-base-1.24.7](#).

--disable-postgresql-sdbc: This switch disables compiling LibreOffice with the ability to connect to a PostgreSQL database. Use it if you have not installed PostgreSQL and you do not want LibreOffice to compile its bundled copy.

--disable-skia: Use [Cairo-1.18.0](#) for rendering instead of skia.

--enable-lto: This switch will enable Link Time Optimization, which results in slightly smaller libraries (about 40 MB). This is said to make LibreOffice programs load faster (and possibly run faster, e.g. when recalculating a spreadsheet). On an 8-core machine with 16 GB of memory, the difference in compile times was minimal, but an extra 2 GB was used for the work files. On machines with fewer processors or a lack of memory, the compile might be much slower.

--without-java: This switch disables Java support in LibreOffice. Java is needed for the deprecated HSQLDB driver, which allows reading databases created by other programs or in earlier versions of libreoffice base. It is also needed for some optional user interface components.

--without-fonts: LibreOffice includes a number of third-party TrueType fonts. If you have already installed some or all of the same fonts, you may prefer to use the system versions.

--enable-evolution2: Enables support for Evolution address books through Evolution Data Server. Note that Evolution Data Server must be installed for this feature to function.

--enable-qt6: This switch enables support for integration with Qt6 and the Qt6 theming engine. If [Qt-6.7.2](#) is not installed in /usr, the include and library directories must be specified in the QT6INC and QT6LIB environment variables.

--enable-kf6: Builds with KDE/Plasma integration. If [Qt-6.7.2](#) and/or [KDE Frameworks-6.5.0](#) are not installed in /usr, the include and library directories must be specified in QT6INC, QT6LIB, KF6INC, and KF6LIB, respectively. This includes a style plugin and KIO integration.

--with-system-abseil: This switch enables using a system version of the Abseil C++ libraries instead of the bundled copy. Add this switch if [Abseil-cpp-20240722.0](#) is installed and you wish to use it instead of the bundled copy.

Contents

Installed Programs: libreoffice, lobase, localc, lodraw, loffice, lofromtemplate, loimpress, lomath, loweb, lowriter, soffice and unopkg; several programs under \$LO_PREFIX/lib/libreoffice/program

Installed Libraries: several libraries under \$LO_PREFIX/lib/libreoffice/program

Installed Directory: \$LO_PREFIX/lib/libreoffice

Short Descriptions

libreoffice (or soffice)	is the main libreoffice suite (symlink to \$LO_PREFIX/lib/libreoffice/program/soffice)
lobase	is the database manager module shell starting script
localc	is the spreadsheet module shell starting script
lodraw	is the vector graphics editor and diagramming tool module shell starting script
loimpress	is the (PowerPoint) presentations editor and display module shell starting script
lomath	is the mathematical formula editor module shell starting script
loweb	is the HTML editor module shell starting script
lowriter	is the word processor module shell starting script
unopkg	is a tool to manage LibreOffice extensions from the command line

Chapter 40. Graphical Web Browsers

This chapter contains a wonderful selection of browsers. We hope you can find one you enjoy using or give them each a trial run.

Epiphany-46.3

Introduction to Epiphany

Epiphany is a simple yet powerful GNOME web browser targeted at non-technical users. Its principles are simplicity and standards compliance.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/epiphany/46/epiphany-46.3.tar.xz>
- Download MD5 sum: 4ecc0fa4dc5dff48c22e1e71b684faaf
- Download size: 3.8 MB
- Estimated disk space required: 47 MB (with tests)
- Estimated build time: 0.3 SBU (with tests; both using parallelism=4)

Epiphany Dependencies

Required

[Gcr-4.3.0](#), [gnome-desktop-44.1](#), [ISO Codes-4.16.0](#), [JSON-GLib-1.8.0](#), [libadwaita-1.5.3](#), [libportal-0.7.1](#), [Nettle-3.10](#), and [WebKitGTK-2.44.3](#) (built with GTK-4)

Optional

[appstream-glib-0.8.3](#) and [Granite](#)

Runtime Dependencies

[gnome-keyring-46.2](#) (for storing passwords) and [Seahorse-43.0](#) (for managing stored passwords)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/epiphany>

Installation of Epiphany

Install Epiphany by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

Now, as the `root` user:

```
ninja install
```

Note

If you installed the package to your system using a “`DESTDIR`” method, `/usr/share/glib-2.0/schemas/gschemas.compiled` was not updated/created. Create (or update) the file using the following command as the `root` user:

```
glib-compile-schemas /usr/share/glib-2.0/schemas
```

One test would fail if this package is not installed, so it's better to run the test suite after installation. To test the results, issue `ninja test`. The tests must be run from a graphical session.

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Program: epiphany

Installed Libraries: None

Installed Directories: /usr/{lib,libexec,share,share/help/*}/epiphany

Short Descriptions

`epiphany` is a GNOME web browser based on the WebKit2 rendering engine

falkon-24.08.0

Introduction to falkon

Falkon is a KDE web browser using the QtWebEngine rendering engine. It was previously known as QuipZilla. It aims to be a lightweight web browser available through all major platforms.

Although falkon is now part of KDE, it can be installed without KDE (with the loss of kwallet functionality).

Falkon relies on QtWebEngine. Please read the warning on that page.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.kde.org/stable/release-service/24.08.0/src/falkon-24.08.0.tar.xz>
- Download MD5 sum: f48e602dfc36152df79287a4945618da
- Download size: 2.5 MB
- Estimated disk space required: 154 MB
- Estimated build time: 1.5 SBU (Using parallelism=4; add 0.9 SBU for tests)

falkon Dependencies

Required

[extra-cmake-modules-6.5.0](#), [KDE Frameworks-6.5.0](#) (for karchive), and [qtwebengine-6.7.2](#)

Note

Strictly speaking, only karchive and ki18n are required to build falkon, but several other packages in [KDE Frameworks-6.5.0](#) can be used if they are present. To build only karchive and ki18n, download those packages from the directory specified in [KDE Frameworks-6.5.0](#) and use the build instructions on that page, changing the \$KF6_PREFIX to /usr.

Optional

[gnome-keyring-46.2](#), [PySide2](#), and [Shiboken2](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/falkon>

Installation of falkon

Install falkon by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      .. &&

make
```

Tests should be run after installation.

Now, as the `root` user:

```
make install &&
ldconfig
```

To test the results, issue: `make test`. All tests should pass. If any fail, the full results will be in `Testing/Temporary/LastTest.log`.

Note

When upgrading this package, it is possible that the contents of the application will remain empty even though everything else appears to be working properly. In this case try removing `~/.cache/falkon` and restarting the program.

If you have installed Pyside2 you will want to examine `hellopython.py` which is in the `scripts/` directory, and perhaps copy it to your home directory.

Note

If you are on a non-x86_64 architecture and run into problems with missing text, you may wish to export `QTWEBENGINE_CHROMIUM_FLAGS="--no-sandbox"` prior to running Falkon.

Command Explanations

`-D BUILD_TESTING=OFF`: This will save a little time and space by not building the test programs, use this if you do not wish to run the test.

Contents

Installed Programs: falkon

Installed Library: libFalkonPrivate.so.3

Installed Directory: /usr/share/falkon

Short Descriptions

`falkon` is a web browser which uses qtwebengine
`libFalkonPrivate.so.3` contains functions used by falkon

Firefox-128.1.0esr

Introduction to Firefox

Firefox is a stand-alone browser based on the Mozilla codebase.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.mozilla.org/pub/firefox/releases/128.1.0esr/source/firefox-128.1.0esr.source.tar.xz>
- Download MD5 sum: efbf349f0b7442fc6f2995bef4d32785
- Download size: 536 MB
- Estimated disk space required: 6.7 GB (223 MB installed) without tests
- Estimated build time: 15 SBU using -j8, without tests

Note

The directory name is firefox-128.1.0

Extracting the tarball will reset the permissions of the current directory to 0755 if you have permission to do that. If you do this in a directory where the sticky bit is set, such as `/tmp` it will end with error messages:

```
tar: ..: Cannot utime: Operation not permitted
tar: ..: Cannot change mode to rwxr-xr-t: Operation not permitted
tar: Exiting with failure status due to previous errors
```

This does finish with non-zero status, but it does *NOT* mean there is a real problem. Do not untar as the `root` user in a directory where the sticky bit is set - that will unset it.

As with other large packages which use C++ (or rust), the SBU times to build this vary more widely than you might expect. The build times will increase significantly if your machine has to swap.

Although upstream prefer to use PulseAudio, for the moment Alsa can still be used. Both may need runtime configuration to get sound working.

Firefox Dependencies

Required

[Cbindgen-0.27.0](#), [GTK+-3.24.43](#), [libnotify-0.8.3](#), [LLVM-18.1.7](#) (with clang, used for bindgen even if using gcc), [nodejs-20.16.0](#), [PulseAudio-17.0](#) (or [alsa-lib-1.2.12](#) if you edit the mozconfig; now deprecated by mozilla), in either case please read the Configuration Information, [Python-3.12.5](#) (after installing [SQLite-3.46.1](#)), [startup-notification-0.12](#), and [UnZip-6.0](#)

Recommended

[ICU-75.1](#), [libevent-2.1.12](#), [libvpx-1.14.1](#), [libwebp-1.4.0](#), [NASM-2.16.03](#), and [nss-3.103](#)

Note

If you don't install recommended dependencies, then internal copies of those packages will be used. They might be tested to work, but they can be out of date or contain security holes.

Optional

[cURL-8.9.1](#), [Doxygen-1.12.0](#), [FFmpeg-7.0.2](#) (runtime, to play mov, mp3 or mp4 files), [GeoClue-2.7.1](#) (runtime), [liboauth-1.0.3](#), [pciutils-3.13.0](#) (runtime), [Valgrind-3.23.0](#), [Wget-1.24.5](#), [Wireless Tools-29](#), [yasm-1.3.0](#), [libproxy](#).

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/firefox>

Installation of Firefox

The configuration of Firefox is accomplished by creating a `mozconfig` file containing the desired configuration options. A default `mozconfig` is created below. To see the entire list of available configuration options (and an abbreviated description of some of them), issue `./mach configure -- --help | less`. You may also wish to review the entire file and uncomment any other desired options. Create the file by issuing the following command:

```
cat > mozconfig << "EOF"
# If you have a multicore machine, all cores will be used by default.

# If you have installed (or will install) wireless-tools, and you wish
# to use geolocation web services, comment out this line
ac_add_options --disable-necko-wifi

# Comment out the following line if you wish not to use Google's Location
# Service (GLS). Note that if Geoclue is installed and configured to use
# GLS (as the BLFS instruction does), Firefox can access GLS via Geoclue
# anyway. On the other hand if Geoclue is not installed (or not properly
# configured) and this line is commented out, the website requiring a
# location service will not function properly.
ac_add_options --with-google-location-service-api-keyfile=$PWD/google-key

# startup-notification is required since firefox-78

# Uncomment the following option if you have not installed PulseAudio and
# want to use alsa instead
#ac_add_options --enable-audio-backends=alsa

# Comment out following options if you have not installed
# recommended dependencies:
ac_add_options --with-system-icu
ac_add_options --with-system-libevent
ac_add_options --with-system-libvpx
ac_add_options --with-system-nspr
ac_add_options --with-system-nss
ac_add_options --with-system-webp

# The libdavd library (av1 decoder) requires nasm. Uncomment this
# if nasm has not been installed. Do not uncomment this if you have
# ffmpeg installed.
#ac_add_options --disable-av1
```

```

# You cannot distribute the binary if you do this.
ac_add_options --enable-official-branding

# Stripping is now enabled by default.
# Uncomment these lines if you need to run a debugger:
#ac_add_options --disable-strip
#ac_add_options --disable-install-strip

# Disabling debug symbols makes the build much smaller and a little
# faster. Comment this if you need to run a debugger.
ac_add_options --disable-debug-symbols

# The BLFS editors recommend not changing anything below this line:
ac_add_options --prefix=/usr
ac_add_options --enable-application=firefox
ac_add_options --enable-crashreporter
ac_add_options --enable-updater

# Enabling the tests will use a lot more space and significantly
# increase the build time, for no obvious benefit.
ac_add_options --disable-tests

# This enables SIMD optimization in the shipped encoding_rs crate.
ac_add_options --enable-rust-simd

ac_add_options --enable-system-ffi
ac_add_options --enable-system-pixman

ac_add_options --with-system-jpeg
ac_add_options --with-system-png
ac_add_options --with-system-zlib

# Sandboxing works well on x86_64 but might cause issues on other
# platforms, e.g. i686.
[ $(uname -m) != x86_64 ] && ac_add_options --disable-sandbox

# Using sandboxed wasm libraries has been moved to all builds instead
# of only mozilla automation builds. It requires extra llvm packages
# and was reported to seriously slow the build. Disable it.
ac_add_options --without-wasm-sandboxed-libraries

# The following option unsets Telemetry Reporting. With the Addons Fiasco,
# Mozilla was found to be collecting user's data, including saved passwords and
# web form data, without users consent. Mozilla was also found shipping updates
# to systems without the user's knowledge or permission.
# As a result of this, use the following command to permanently disable
# telemetry reporting in Firefox.
unset MOZ_TELEMETRY_REPORTING

mk_add_options MOZ_OBJDIR=@TOPSRCDIR@/firefox-build-dir
EOF

```

Adapt two files for compatibility with cbindgen-0.27.0 or later:

```

sed 's/input.try/&_parse/' \
    -i servo/components/style_traits/values.rs &&
sed '0,/"Keyframe"/{//d}' -i servo/ports/geckolib/cbindgen.toml

```

If building with system ICU, adapt the line break mapping for ICU 74 or later. This construct adds five entries to a c++ array.

```

for i in {43..47}; do
    sed -i '/ZWJ/s/,CLASS_CHARACTER/` intl/lwbrk/LineBreaker.cpp
done

```

Compile Firefox by issuing the following commands:

If the geolocation APIs are needed:

Note

The Google API Key below is specific to LFS. If using these instructions for another distro, or if you intend to distribute binary copies of the software using these instructions, please obtain your own key following the

instructions located at <https://www.chromium.org/developers/how-tos/api-keys>.

```
echo "AIzaSyDxKL42zsPjbke508_rPVpVrLrJ8aeE9rQ" > google-key
```

Note

If you are compiling this package in chroot you must ensure that `/dev/shm` is mounted. If you do not do this, the Python configuration will fail with a traceback report referencing `/usr/lib/pythonN.N/multiprocessing/synchronize.py`. As the `root` user, run:

```
mountpoint -q /dev/shm || mount -t tmpfs devshm /dev/shm
```

Now invoke the Python `mach` script to compile the package.

```
export MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE=none &&
export MOZBUILD_STATE_PATH=${PWD}/mozbuild &&
./mach build
```

The `mozconfig` above disables the tests because they use a lot more time and disk space for no obvious benefit. If you have nevertheless enabled them, you can run the tests by executing `./mach gtest`. This will require a network connection, and to be run from within an Xorg session - there is a popup dialog when it fails to connect to ALSA (that does not create a failed test). One or two tests will fail. To see the details of the failure(s) you will need to log the output from that command so that you can review it.

Now, as the `root` user:

```
export MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE=none &&
./mach install
```

Empty the environment variables which were set above:

```
unset MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE
unset MOZBUILD_STATE_PATH
```

Command Explanations

`export MOZBUILD_STATE_PATH=${PWD}/mozbuild`: The build tells you that it intends to create `~/.mozbuild`, and offer you an option to press <ENTER> to accept this, or Ctrl-C to cancel and restart the build after specifying the directory. In practice, the message may not appear until after <ENTER> is keyed, i.e. the build stalls.

That directory is used for a (probably random) telemetry identifier. Creating this directory within the build directory and deleting it after the installation prevents it being used.

`MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE=none`: Use the system python to create a virtual environment for `mach` without downloading any python wheels and without using the system python modules. This prevents version mismatches between system modules and bundled ones.

`./mach build --verbose`: Use this alternative if you need details of which files are being compiled, together with any C or C++ flags being used. But do not add '--verbose' to the install command since it is not accepted there.

`./mach build -jN`: The build should, by default, use all the online CPU cores. If using all the cores causes the build to swap because you have insufficient memory, using fewer cores can be faster.

`CC=gcc CXX=g++`: BLFS used to prefer to use gcc and g++ instead of upstream's defaults of the clang programs. With the release of gcc-12 the build takes longer with gcc and g++, primarily because of extra warnings, and is bigger. Set these environment variables *before you run the configure script* if you wish to continue to use gcc, g++. Building with GCC on i? 86 is currently broken.

Configuring Firefox

If you use a desktop environment like Gnome or KDE you may want to create a `firefox.desktop` file so that Firefox appears in the panel's menus. As the `root` user:

```
mkdir -pv /usr/share/applications &&
mkdir -pv /usr/share/pixmaps &&
MIMETYPE="text/xml;text/mml;text/html;" &&
MIMETYPE+="application/xhtml+xml;application/vnd.mozilla.xul+xml;" &&
```

```

MIMETYPE+="x-scheme-handler/http;x-scheme-handler/https"      &&

cat > /usr/share/applications/firefox.desktop << EOF &&
[Desktop Entry]
Encoding=UTF-8
Name=Firefox Web Browser
Comment=Browse the World Wide Web
GenericName=Web Browser
Exec=firefox %u
Terminal=false
Type=Application
Icon=firefox
Categories=GNOME;GTK;Network;WebBrowser;
MimeType=$MIMETYPE
StartupNotify=true
EOF

unset MIMETYPE &&

ln -sfv /usr/lib/firefox/browser/chrome/icons/default/default128.png \
/usr/share/pixmaps/firefox.png

```

Configuration Information

The application settings for firefox are accessible by keying `about:config` in the address bar.

Occasionally, getting working sound in firefox can be a problem. Although upstream prefers pulseaudio, on balance using Alsa may be easier.

If you enabled Alsa for sound, you may need to alter one variable to get working sound. If you run `firefox` from a terminal and try to play something with sound you might encounter error messages like:

```
Sandbox: seccomp sandbox violation: pid 3941, tid 4030, syscall 16, args 48 2147767296 139909894784796 0 0 0.
```

That was on x86_64, on i686 the syscall number is 54. To allow this syscall, in `about:config` change `security.sandbox.content.syscall_whitelist` to 16 (or 54 if using i686).

If you use `pulseaudio` in a Desktop Environment, it might already be started by that DE. But if it is not, although firefox-57 managed to start it, firefox-58 did not. If you run `firefox` from a terminal and this problem is present, trying to play sound will encounter error messages warning `Can't get cubeb context!`

The fix for this is to close firefox, start pulseaudio to check it does start (if not, read the information on Configuring in [PulseAudio-17.0](#)) and restart firefox to check it is working. If it now works, add the following to your `~/.xinitrc: pulseaudio -verbose --log-target=journald&` (unfortunately, on some systems this does not work).

You may wish to use multiple profiles within firefox. To do that, invoke firefox as `firefox --ProfileManager`. You can also check which profile is currently in use from `about:profiles`.

Although WebRender (using the GPU for compositing) is not used by default, it now appears to work well on supported hardware (ATI, Nvidia and Intel GPUs with Mesa-18 or later). For an explanation, please see [hacks.mozilla.org](#). The only downside seems to be that on a machine with limited RAM it might use more RAM.

To check if WebRender is being used, look in `about:support`. In the Graphics section, Compositing will either show 'Basic' (i.e. not in use) or 'WebRender'. To enable it, go to `about:config` and change `gfx.webrender.all` to True. You will need to restart firefox.

It may be useful to mention the processes from firefox which can appear in `top` - as well as firefox itself, there may be multiple Web Content processes, and now an RDD Process (Remote Data Decoder) which appears when playing web videos encoded with av1 (libdav1d). If WebRender has been enabled, a GPU Process will also appear when firefox has to repaint (e.g. scrolling, opening a new tab, or playing a video).

Contents

Installed Programs: `firefox`

Installed Libraries: Numerous libraries, browser components, plugins, extensions, and helper modules installed in `/usr/lib/firefox`

Installed Directory: `/usr/lib/firefox`

Short Descriptions

`firefox` is a GTK+-3 internet browser that uses the Mozilla Gecko rendering engine

Seamonkey-2.53.18.2

Introduction to Seamonkey

Seamonkey is a browser suite, a descendant of Netscape. It includes the browser, composer, mail and news clients, and an IRC client.

It is the community-driven follow-on to the Mozilla Application Suite, created after Mozilla decided to focus on separate applications for browsing and e-mail. Those applications are [Firefox-128.1.0](#) and [Thunderbird-128.1.0esr](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.seamonkey-project.org/releases/2.53.18.2/source/seamonkey-2.53.18.2.source.tar.xz>
- Download MD5 sum: 266a86651348e21934707a7ccb3abef3
- Download size: 241 MB
- Estimated disk space required: 3.0 GB (148 MB installed)
- Estimated build time: 3.5 SBU (with parallelism=8)

Additional Downloads

- Recommended patch (required for building with system ICU-75 or newer):
<https://www.linuxfromscratch.org/patches/blfs/12.2/seamonkey-2.53.18.2-cxx17-1.patch>

Note

The tarball *seamonkey-2.53.18.2.source.tar.xz* will untar to *seamonkey-2.53.18.2* directory.

Seamonkey Dependencies

Required

[Cbindgen-0.27.0](#), [GTK+-3.24.43](#), [nodejs-20.16.0](#), [Python-3.11.1](#), [UnZip-6.0](#), [yasm-1.3.0](#), and [Zip-3.0](#)

Recommended

[ICU-75.1](#), [libevent-2.1.12](#), [libwebp-1.4.0](#), [LLVM-18.1.7](#) (with clang), [NASM-2.16.03](#), [NSPR-4.35](#), [NSS-3.103](#), and [PulseAudio-17.0](#)

Note

If you don't install recommended dependencies, then internal copies of those packages will be used. They might be tested to work, but they can be out of date or contain security holes.

Optional

[alsa-lib-1.2.12](#), [dbus-glib-0.112](#), [startup-notification-0.12](#), [Valgrind-3.23.0](#), [Wget-1.24.5](#), [Wireless Tools-29](#), [Hunspell](#), [Gconf](#), and [Watchman](#)

Installation of Seamonkey

The configuration of Seamonkey is accomplished by creating a `mozconfig` file containing the desired configuration options. A default `mozconfig` file is created below. To see the entire list of available configuration options (and an abbreviated description of each one), issue `./configure --help`. You may also wish to review the entire file and uncomment any other desired options. Create the file by issuing the following command:

```
cat > mozconfig << "EOF"
# If you have a multicore machine, all cores will be used

# If you have installed DBus-Glib comment out this line:
```

```
ac_add_options --disable-dbus

# If you have installed dbus-glib, and you have installed (or will install)
# wireless-tools, and you wish to use geolocation web services, comment out
# this line
ac_add_options --disable-necko-wifi

# Uncomment these lines if you have installed optional dependencies:
#ac_add_options --enable-system-hunspell
#ac_add_options --enable-startup-notification

# Uncomment the following option if you have not installed PulseAudio
#ac_add_options --disable-pulseaudio
# and uncomment this if you installed alsa-lib instead of PulseAudio
#ac_add_options --enable-alsa

# Comment out the following option if you have gconf installed
ac_add_options --disable-gconf

# Comment out following options if you have not installed
# recommended dependencies:
ac_add_options --with-system-icu
ac_add_options --with-system-libevent
ac_add_options --with-system-nspr
ac_add_options --with-system-nss
ac_add_options --with-system-webp

# Disabling debug symbols makes the build much smaller and a little
# faster. Comment this if you need to run a debugger.
ac_add_options --disable-debug-symbols

# The elf-hack is reported to cause failed installs (after successful builds)
# on some machines. It is supposed to improve startup time and it shrinks
# libxul.so by a few MB. With recent Binutils releases the linker already
# supports a much safer and generic way for this.
ac_add_options --disable-elf-hack
ac_add_options --enable-linker=bfd
export LDFLAGS="$LDFLAGS -Wl,-z,pack-relative-relocs"

# Seamonkey has some additional features that are not turned on by default,
# such as an IRC client, calendar, and DOM Inspector. The DOM Inspector
# aids with designing web pages. Comment these options if you do not
# desire these features.
ac_add_options --enable-calendar
ac_add_options --enable-dominspector
ac_add_options --enable-irc

# The BLFS editors recommend not changing anything below this line:
ac_add_options --prefix=/usr
ac_add_options --enable-application=comm/suite

ac_add_options --disable-crashreporter
ac_add_options --disable-updater
ac_add_options --disable-tests

# The SIMD code relies on the unmaintained packed_simd crate which
# fails to build with Rustc >= 1.78.0. We may re-enable it once
# Mozilla port the code to use std::simd and std::simd is stabilized.
ac_add_options --disable-rust-simd

ac_add_options --enable-strip
ac_add_options --enable-install-strip

# You cannot distribute the binary if you do this.
ac_add_options --enable-official-branding

# The option to use system cairo was removed in 2.53.9.
ac_add_options --enable-system-ffi
ac_add_options --enable-system-pixman

ac_add_options --with-system-bz2
ac_add_options --with-system-jpeg
ac_add_options --with-system-png
ac_add_options --with-system-zlib
```

```
export CC=clang CXX=clang++
EOF
```

Note

If you are compiling this package in chroot you must ensure that `/dev/shm` is mounted. If you do not do this, the Python configuration will fail with a traceback report referencing `/usr/lib/pythonN.N/multiprocessing/synchronize.py`. As the `root` user, run:

```
mountpoint -q /dev/shm || mount -t tmpfs devshm /dev/shm
```

First, if you are building with system ICU, adapt the line break mapping for ICU-74 or later, and apply a patch to build this package with the C++17 standard because the headers of ICU-75 or later require some C++17 features:

```
(for i in {43..47}; do
    sed '/ZWJ/s/$/,CLASS_CHARACTER/' -i intl/lwbrk/LineBreaker.cpp || exit $?
done) &&

patch -Np1 -i ../seamonkey-2.53.18.2-cxx17-1.patch
```

Next, fix a problem with the bundled 'distro' python module:

```
sed -e '1012 s/stderr=devnull/stderr=subprocess.DEVNULL/' \
      -e '1013 s/OSError/(OSError, subprocess.CalledProcessError)/' \
      -i third_party/python/distro/distro.py
```

Compile Seamonkey by running the following commands:

```
export PATH_PY311=/opt/python3.11/bin:$PATH &&
PATH=$PATH_PY311 AUTOCONF=true ./mach build
```

This package does not come with a test suite.

Install Seamonkey by issuing the following commands as the `root` user:

Note

If `sudo` or `su` is invoked for opening a shell running as the `root` user, ensure `PATH_PY311` is correctly passed or the following command will fail. For `sudo`, use the `--preserve-env=PATH_PY311` option. For `su`, do *not* use the `-` or `--login` options.

```
PATH=$PATH_PY311 ./mach install &&
chown -R 0:0 /usr/lib/seamonkey &&

cp -v $(find -name seamonkey.1 | head -n1) /usr/share/man/man1
```

Finally, unset the `PATH_PY311` variable:

```
unset $PATH_PY311
```

Command Explanations

`export CC=clang CXX=clang++`: With the introduction of gcc-12, many more warnings are generated when compiling mozilla applications and that results in a much slower, and larger, build. Furthermore, building with GCC on i?86 is currently broken. Although upstream mozilla code defaults to using llvm unless overridden, the older configure code in Seamonkey defaults to gcc.

`./mach build --verbose`: Use this alternative if you need details of which files are being compiled, together with any C or C++ flags being used. But do not add '--verbose' to the install command, it is not accepted there.

`./mach build -jN`: The build should, by default, use all the online CPU cores. If using all the cores causes the build to swap because you have insufficient memory, using fewer cores can be faster.

Configuring Seamonkey

For installing various Seamonkey add-ons, refer to [Add-ons for Seamonkey](#).

Along with using the "Preferences" menu to configure Seamonkey's options and preferences to suit individual tastes, finer grain control of many options is only available using a tool not available from the general menu system. To access this tool, you'll need to open a browser window and enter `about:config` in the address bar. This will display a list of the configuration preferences and information related to each one. You can use the "Search:" bar to enter search criteria and narrow down the listed items. Changing a preference can be done using two methods. One, if the preference has a boolean value (True/False), simply double-click on the preference to toggle the value and two, for other preferences simply right-click on the desired line, choose "Modify" from the menu and change the value. Creating new preference items is accomplished in the same way, except choose "New" from the menu and provide the desired data into the fields when prompted.

If you use a desktop environment like Gnome or KDE you may wish to create a `seamonkey.desktop` file so that Seamonkey appears in the panel's menus. If you didn't enable Startup-Notification in your mozconfig change the StartupNotify line to false. As the `root` user:

```
mkdir -pv /usr/share/{applications,pixmaps} &&  
  
cat > /usr/share/applications/seamonkey.desktop << "EOF"  
[Desktop Entry]  
Encoding=UTF-8  
Type=Application  
Name=Seamonkey  
Comment=The Mozilla Suite  
Icon=seamonkey  
Exec=seamonkey  
Categories=Network;GTK;Application;Email;Browser;WebBrowser;News;  
StartupNotify=true  
Terminal=false  
EOF  
  
ln -sfv /usr/lib/seamonkey/chrome/icons/default/default128.png \  
/usr/share/pixmaps/seamonkey.png
```

Contents

Installed Programs: `seamonkey`

Installed Libraries: Numerous libraries, browser, and email/newsgroup components, plugins, extensions, and helper modules installed in `/usr/lib/seamonkey`

Installed Directory: `/usr/lib/seamonkey`

Short Descriptions

`seamonkey` is the Mozilla browser/email/newsgroup/chat client suite

Chapter 41. Other X-based Programs

These programs use the X Window System and don't fit easily into any of the other chapters.

Balsa-2.6.4

Introduction to Balsa

The Balsa package contains a GNOME-2 based mail client.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://pawsa.fedorapeople.org/balsa/balsa-2.6.4.tar.xz>
- Download MD5 sum: `beb22cb6bac87ee05255c48ee1d28dd`
- Download size: 3.5 MB
- Estimated disk space required: 66 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/balsa-2.6.4-upstream_fixes-2.patch

Balsa Dependencies

Required

[Aspell-0.60.8.1](#), [enchant-2.8.2](#), [GMime-3.2.7](#), [GPGME-1.23.2](#), [GTK+-3.24.43](#), and [libnotify-0.8.3](#)

Required (Runtime)

MTA (that provides a `sendmail` command)

Optional

[Compface-1.5.2](#), [Gcr-3.41.2](#), [gtksourceview-3.24.11](#), [libcanberra-0.30](#), [libsecret-0.21.4](#), [MIT Kerberos V5-1.21.3](#), [OpenLDAP-2.6.8](#), [Procmail-3.24](#), [SQLite-3.46.1](#), [WebKitGTK-2.44.3](#), [html2text](#), [osmo](#), [rubrica](#), [yelp-tools](#)

Installation of Balsa

First, fix build issues when configuring without WebKitGTK, or when using WebKitGTK-2.38 or later.

```
patch -Np1 -i ../../balsa-2.6.4-upstream_fixes-2.patch
```

Install Balsa by running the following commands:

```
./configure --prefix=/usr \
           --sysconfdir=/etc \
           --localstatedir=/var/lib \
           --without-html-widget &&
make
```

This package does not come with a test suite.

Now, as the *root* user:

make install

Command Explanations

`--without-html-widget`: This option disables building the HTML renderer. If you want the HTML renderer to be built, verify that you have [WebKitGTK-2.44.3](#) installed, and then remove this option.

--with-canberra: Enable libcanberra support.

--with-compface: Enable compface support.

--with-gcr: Use libgcr for TLS support.

--with-gss: Enable GSSAPI support.

--with-ldap: Use this option to enable LDAP address book support if OpenLDAP is installed.

--with-libsecret: Enable strong password support and support for gnome-keyring to store passwords.

--with-sqlite: Use this option to enable SOLite address book support if SOLite is installed.

Contents

Installed Programs: balsa and balsa-ab

Installed Libraries: None

Installed Directories: /etc/sounds/events, /usr/share/balsa, and /usr/share/sounds/balsa

Short Descriptions

balsa is a glib based mail client
balsa-ab is the address-book application used by **balsa**

Introduction to feh

feh is a fast, lightweight image viewer which uses Imlib2. It is commandline-driven and supports multiple images through slideshows, thumbnail browsing or multiple windows, and montages or index prints (using TrueType fonts to display file info). Advanced features include fast dynamic zooming, progressive loading, loading via HTTP (with reload support for watching webcams), recursive file opening (slideshow of a directory hierarchy), and mouse wheel/keyboard control.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://feh.finalrewind.org/feh-3.10.3.tar.bz2>
- Download MD5 sum: 4f9e2c6ecc120852ba86dc41df366aba
- Download size: 2.0 MB
- Estimated disk space required: 5.2 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

feh Dependencies

Required

[libpng-1.6.43](#) and [imlib2-1.12.3](#) (built with [giflib-5.2.2](#) support, for the tests)

Recommended

[cURL-8.9.1](#)

Optional

[libexif-0.6.24](#)

Optional (runtime)

[libjpeg-turbo-3.0.1](#) (for lossless image rotation) and [ImageMagick-7.1.1-36](#) (to load unsupported formats)

Optional (test suite)

[Test-Command-0.11](#) (required) and [mandoc-1.14.6](#) (optional)

Installation of feh

Install feh by running the following commands:

```
sed -i "s:doc/feh:&-3.10.3:" config.mk &&
make PREFIX=/usr
```

To test the results, issue: `make test`.

Now, as the `root` user:

```
make PREFIX=/usr install
```

Command Explanations

`sed -i "s:doc/feh:&-3.10.3:" config.mk`: This sed fixes the doc directory to a versioned one, as used in BLFS.

`curl=0`: Use this make flag if you don't have the CURL package installed.

`exif=1`: This make flag enables builtin Exif tag display support.

Contents

Installed Program: feh

Installed Libraries: None

Installed Directories: /usr/share/doc/feh-3.10.3 and /usr/share/feh

Short Descriptions

`feh` is an image viewer and cataloguer

FontForge-20230101

Introduction to FontForge

The FontForge package contains an outline font editor that lets you create your own postscript, truetype, opentype, cid-keyed, multi-master, cff, svg and bitmap (bdf, FON, NFNT) fonts, or edit existing ones.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/fontforge/fontforge/releases/download/20230101/fontforge-20230101.tar.xz>
- Download MD5 sum: 7043f25368ed25bcd75d168564919fb7
- Download size: 13 MB
- Estimated disk space required: 102 MB (add 52 MB for the tests)
- Estimated build time: 0.4 SBU (with parallelism=4, add 0.1 SBU for the tests)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/fontforge-20230101-security_fixes-1.patch

FontForge Dependencies

Required

[libspiro-20220722](#) and [libxml2-2.13.3](#)

Recommended

[GTK+-3.24.43](#)

Optional

[glib-5.2.2](#), [git-2.46.0](#), [libjpeg-turbo-3.0.1](#), [libtiff-4.6.0](#), [sphinx-8.0.2](#) (to build html documentation), and [WOFF2-1.0.2](#)

Installation of FontForge

First, fix two security vulnerabilities in the Splinefont functionality:

```
patch -Np1 -i ../../fontforge-20230101-security_fixes-1.patch
```

Next, fix a problem with old translations exposed by gettext-0.22:

```
sed -i.orig 's/^\(%[^%[:space:]]*\)\hs/\1s/g' \
fontforgeexe/searchview.c po/de.po po/ca.po \
po/hr.po po/it.po po/pl.po po/uk.po po/en_GB.po \
po/fr.po po/vi.po po/ko.po po/ja.po
```

Install FontForge by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
-D CMAKE_BUILD_TYPE=Release \
-W no-dev .. &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

If you have installed `sphinx-build`, HTML documentation was built and installed in `/usr/share/doc/fontforge`. To be able to access it as a versioned directory, create a symlink as the `root` user:

```
ln -sv fontforge /usr/share/doc/fontforge-20230101
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to create an optimized release without debug information.

`-W no-dev`: This switch is used to suppress warnings intended for the package's developers.

`-D ENABLE_X11=ON`: use an X11 backend instead of GDK3 (gtk+-3) for basic Xorg graphics.

Contents

Installed Programs: fontforge, fontimage, fontlint and sfddiff

Installed Library: libfontforge.so and /usr/lib/python3.12.5/site-packages/{fontforge,psMat}.so

Installed Directories: /usr/share/fontforge and optionally /usr/share/doc/fontforge{-,20230101}

Short Descriptions

- | | |
|------------------------|---|
| <code>fontforge</code> | is a program that allows you to create and modify font files |
| <code>fontimage</code> | is a program that produces an image showing representative glyphs of the font |
| <code>fontlint</code> | is a program that checks the font for certain common errors |
| <code>sfddiff</code> | is a program that compares two font files |

FreeRDP-3.7.0

Introduction to FreeRDP

The FreeRDP package contains libraries and utilities for utilizing the Remote Desktop Protocol. This includes tools to run an RDP server as well as connecting to a computer using RDP. This is primarily used for connecting to Microsoft Windows computers, but can also be used on Linux and macOS.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/freerdp/freerdp/archive/3.7.0/FreeRDP-3.7.0.tar.gz>
- Download MD5 sum: 4658086473bb78ab6b1d876efe454e12
- Download size: 9.8 MB
- Estimated disk space required: 131 MB
- Estimated build time: 0.6 SBU

FreeRDP Dependencies

Required

[FFmpeg-7.0.2](#), [ICU-75.1](#), and [Xorg Libraries](#)

Recommended

[Cairo-1.18.0](#), [docbook-xsl-nons-1.79.2](#), [Fuse-3.16.2](#), [JSON-C-0.17](#), [MIT Kerberos V5-1.21.3](#), [libusb-1.0.27](#), [libxkbcommon-1.7.0](#), and [Wayland-1.23.0](#)

Optional

[Cups-2.4.10](#), [FAAC-1_30](#), [FAAD2-2.11.1](#), [fdk-aac-2.0.3](#), [FreeRDP-3.7.0](#), [LAME-3.100](#), [Linux-PAM-1.6.1](#), [PulseAudio-17.0](#), [cJSON](#), [GSM](#), [ocl-icd](#), [mbedtls](#), [openh264](#), [pcsc-lite](#), [SDL_ttf](#), [soxr](#), and [uriparser](#)

Installation of FreeRDP

Install FreeRDP by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr      \
-D CMAKE_SKIP_INSTALL_RPATH=ON        \
-D CMAKE_BUILD_TYPE=Release          \
-D WITH_CAIRO=ON                   \
-D WITH_CLIENT_SDL=OFF              \
-D WITH_DSP_FFMPEG=ON               \
-D WITH_FFMPEG=ON                  \
-D WITH_PCSC=OFF                   \
-D WITH_SERVER=ON                  \
-D WITH_SERVER_CHANNELS=ON          \
-D DOCBOOKXSL_DIR=/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2 \
-W no-dev                          \
-G Ninja ..                         &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D CMAKE_SKIP_INSTALL_RPATH=ON`: This switch makes `cmake` remove hardcoded library search paths (rpath) when installing a binary executable file or a shared library. This package does not need rpath once it's installed into the standard location, and rpath may sometimes cause unwanted effects or even security issues.

`-D DOCBOOKXSL_DIR=/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2`: This parameter points CMake to the location of the Docbook XSL stylesheets, and is needed because BLFS does not use namespaced versions of the stylesheets. This is needed to create man pages. If you do not want manpages, pass `-D WITH_MANPAGES=OFF` to CMake instead.

`-D WITH_CAIRO=ON`: This parameter uses Cairo for HighDPI and window resizing support.

`-D WITH_CLIENT_SDL=OFF`: This parameter disables using SDL to draw the window when using the FreeRDP client. This is disabled because it depends on [SDL_ttf](#). Remove this switch if you have the `SDL_ttf` package installed.

`-D WITH_FFMPEG=ON`: This parameter enables support for using ffmpeg for the H.264 graphics rendering mode, which is required when connecting via RDP to Windows Server 2012 (or Windows 8) or later hosts.

`-D WITH_DSP_FFMPEG=ON`: This parameter enables support for using ffmpeg for sound and microphone redirection.

`-D WITH_PCSC=OFF`: This parameter disables support for SmartCards since BLFS does not have the required dependency for it.

`-D WITH_SERVER=ON`: This parameter enables building the FreeRDP server components. Remove this parameter if you do not want to use the FreeRDP server.

`-D WITH_SERVER_CHANNELS=ON`: This parameter builds additional plugins for the FreeRDP server.

`-D WITH_LAME=ON`: Use this option if you wish to enable support for using LAME to provide MP3 Audio Codec support.

`-D WITH_FAAC=ON`: Use this option if you wish to enable support for using FAAC to provide AAC Audio Codec support.

`-D WITH_FAAD2=ON`: Use this option if you wish to enable support for using FAAD2 to provide AAC Audio Codec support.

`-D WITH_FDK_AAC=ON`: Use this option if you wish to enable support for using fdk-aac to provide AAC Audio Codec support.

`-D WITHMBEDTLS=ON`: Use this option if you have the optional dependency [mbedtls](#) installed and wish to use it as an alternative to OpenSSL for cryptography support.

`-D WITH_OPENCL=ON`: Use this option if you have the optional dependency [ocl-icd](#) installed and wish to use OpenCL support.

`-D WITH_OPENH264=ON`: Use this option if you have the optional dependency [openh264](#) installed and wish to use it for H.264 support instead of ffmpeg.

`-D WITH_GSM=ON`: Use this option if you have the optional dependency [GSM](#) installed and wish to use GSM as an audio codec.

`-D WITH_SOXR=ON`: Use this option if you have the optional dependency [soxr](#) installed and wish to use it instead of ffmpeg for multimedia redirection.

Contents

Installed Programs: freerdp-proxy, freerdp-shadow-cli, sfreerdp, sfreerdp-server, winpr-hash, winpr-makecert, wlfreerdp, xfreerdp

Installed Libraries: libfreerdp3.so, libfreerdp-client3.so, libfreerdp-server3.so, libfreerdp-server-proxy3.so, libfreerdp-shadow3.so, libfreerdp-shadow-subsystem3.so, librtk0.so, libuwac0.so, libwinpr3.so, and libwinpr-tools3.so

Installed Directories: /usr/include/{freerdp3,rdtk0,uwac0,winpr3}, /usr/lib/cmake/{FreeRDP3,FreeRDP-Client3,FreeRDP-Proxy3,FreeRDP-Server3,rdtk0,uwac0,WinPR3,WinPR-tools3}, /usr/lib/freerdp3, /usr/share/FreeRDP

Short Descriptions

<code>freerdp-proxy</code>	configures FreeRDP to use a proxy server
<code>freerdp-shadow-cli</code>	interfaces with the shadow functionality in FreeRDP
<code>sfreerdp</code>	is a utility for sharing an X display over RDP
<code>sfreerdp-server</code>	is a utility for sharing an X display over RDP
<code>winpr-hash</code>	creates a NTLM hash from a username and password pair
<code>winpr-makecert</code>	creates X.509 certificates for use with FreeRDP
<code>wlfreerdp</code>	is a RDP client for Wayland
<code>xfreerdp</code>	is a RDP client for X11
<code>libfreerdp3.so</code>	contains functions that provide RDP client and server functionality

Gimp-20240711

Introduction to Gimp

The Gimp package contains the GNU Image Manipulation Program which is useful for photo retouching, image composition and image authoring.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/gimp/gimp3-20240711.tar.xz>
- Download MD5 sum: 17fc0bfc8e66fc61d672b4409cb3e3a
- Download size: 26 MB
- Estimated disk space required: 456 MB (162 MB installed)
- Estimated build time: 1.2 SBU (Using parallelism=4)

Additional Downloads

Note

The English version of the help files is complete, but large parts of the text for many other languages are not yet translated.

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/gimp/gimp3-help-20240711.tar.xz>
- Download size: 129 MB
- Download MD5 sum: 162d57a67ff737ed221a8d7f274d119f
- Estimated disk space required: From 351 MB (74 MB installed) for en only to 1.5 GB for all languages
- Estimated build time: 0.4 SBU (en only), 5.1 SBU for all languages; using parallelism=4

Gimp Dependencies

Required

[appstream-glib-0.8.3](#), [gegl-0.4.48](#), [gexiv2-0.14.3](#), [glib-networking-2.80.0](#), [GTK+-3.24.43](#), [harfBuzz-9.0.0](#), [libmypaint-1.6.1](#), [librsvg-2.58.3](#), [libtiff-4.6.0](#), [libxml2-2.13.3](#) (to build the translated help files), [Little CMS-2.16](#), [mypaint-brushes-1.3.1](#), and [Poppler-24.08.0](#) (including poppler-data)

Recommended

[Graphviz-12.1.0](#), [ghostscript-10.03.1](#) (with libgs installed), [Gvfs-1.54.2](#) and [glib-networking-2.80.0](#) (both runtime, to access the online help, open images from a HTTP or HTTPS URL, drag and drop images from an external application, etc.), [ISO Codes-4.16.0](#), [libgudev-238](#), [PyGObject-3.48.2](#), and [xdg-utils-1.2.1](#)

Optional

[AAlib-1.4rc5](#), [alsa-lib-1.2.12](#), [Gjs-1.80.2](#), [libjxl-0.10.3](#), [libmng-2.0.3](#), [libunwind-1.6.2](#), [libwebp-1.4.0](#), [Lua-5.4.7](#), [OpenJPEG-2.5.2](#), [GTK-Doc-1.34.0](#), [cfitsio](#), [libbacktrace](#), [libiff](#), [libilbm](#), [libheif](#) with [libde265](#) (both needed to read macOS heic images), [libwmf](#), [OpenEXR](#), and [qoi](#)

Optional, for building the help system

[dblatex](#) (for PDF docs), [pngnq](#) and [pngcrush](#) to optimize the png files, but see the note on the help download above

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/gimp>

Installation of Gimp

Install Gimp by running the following commands:

```
mkdir gimp-build &&
cd gimp-build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

To test the results (requires an X-Windowed terminal) issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+-3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Installation of Gimp-Help

The `gimp-help` tarball contains images and English text help for help files, together with translations. If you wish to install local copies of the help files to read offline, unpack the `gimp-help` tarball and change into the root of the newly created source tree.

```
tar -xf ../../gimp3-help-20240711.tar.xz &&
cd gimp3-help-20240711

automake --add-missing

ALL_LINGUAS="ca cs da de en en_GB es fa fr hr ko lt nl nn pt pt_BR ro sl sv uk zh_CN" \
./configure --prefix=/usr
```

Remove from `ALL_LINGUAS` the codes for any languages which you do not wish to install. Alternatively, remove the line starting with `ALL_LINGUAS`, if you wish to build all languages.

Now build the help files:

```
make
```

Issue the following commands as the `root` user to install the help files:

```
make install &&
chown -R root:root /usr/share/gimp/3.0/help
```

Command Explanations

`ALL_LINGUAS="ca cs da de en en_GB es fa...":` by default, the help files will be rendered in all the available languages. Remove the codes of any languages you do not wish to build.

Configuring The Gimp

Config Files

```
/etc/gimp/3.0/*
```

Contents

Installed Programs: gimp-2.99, gimp-console-2.99, gimptool-2.99, gimp-test-clipboard-2.99, and gimp-script-fu-interpreter-3.0

Installed Libraries: libgimp-3.0.so, libgimp-scriptfu-3.0, libgimpbase-3.0.so, libgimpcolor-3.0.so, libgimpconfig-3.0.so, libgimpmath-3.0.so, libgimpmodule-3.0.so, libgimpui-3.0.so, and libgimpwidgets-3.0.so

Installed Directories: /etc/gimp, /usr/include/gimp-3.0, /usr/lib/gimp-3.0, /usr/lib/gimp/3.0, /usr/share/gimp/3.0, and /usr/share/doc/gimp-2.99

Short Descriptions

<code>gimp-2.99</code>	is the Gnu Image Manipulation Program. It works with a variety of image formats and provides a large selection of tools
<code>gimp-console-2.99</code>	is a console program that behaves as if The Gimp was called with the <code>--no-interface</code> command-line option
<code>gimptool-2.99</code>	is a tool that can build plug-ins or scripts and install them if they are distributed in one source file. <code>gimptool-2.99</code> can also be used by programs that need to know what libraries and include-paths The Gimp was compiled with
<code>libgimp-3.0.so</code>	provides C bindings for The Gimp's Procedural Database (PDB), which offers an interface to core functions and to functionality provided by plug-ins
<code>libgimpbase-3.0.so</code>	provides the C functions for basic Gimp functionality such as determining enumeration data types, gettext translation, determining The Gimp's version number and capabilities, handling data files and accessing the environment
<code>libgimpcolor-3.0.so</code>	provides the C functions relating to RGB, HSV and CMYK colors as well as converting colors between different color models and performing adaptive supersampling on an area
<code>libgimpconfig-3.0.so</code>	contains C functions for reading and writing config information
<code>libgimpmath-3.0.so</code>	contains C functions which provide mathematical definitions and macros, manipulate 3x3 transformation matrices, set up and manipulate vectors and the MD5 message-digest algorithm
<code>libgimpmodule-3.0.so</code>	provides the C functions which implement module loading using GModule and supports keeping a list of GimpModule's found in a given searchpath
<code>libgimpthumb-3.0.so</code>	provides the C functions for handling The Gimp's thumbnail objects
<code>libgimpui-3.0.so</code>	contains The Gimp's common user interface functions
<code>libgimpwidgets-3.0.so</code>	contains The Gimp and GTK's widget creation and manipulation functions

Gparted-1.6.0

Introduction to Gparted

Gparted is the Gnome Partition Editor, a Gtk 3 GUI for other command line tools that can create, reorganise or delete disk partitions.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/gparted/gparted-1.6.0.tar.gz>
- Download MD5 sum: b2006a0a3f35853e7d7dc34c87db11f2
- Download size: 5.2 MB
- Estimated disk space required: 72 MB (add 70 MB for tests)
- Estimated build time: 0.3 SBU (using parallelism=4; add 0.3 SBU for tests)

Gparted Dependencies

Required

[Gtkmm-3.24.9](#) and [parted-3.6](#)

Optional

[btrfs-progs-6.10.1](#) (if using a btrfs filesystem), [exfatprogs](#), and [udftools](#)

Installation of Gparted

Install Gparted by running the following commands:

```
./configure --prefix=/usr \
            \
            \
            \
make
```

To run the tests, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-doc`: This switch disables building of the optional documentation. Remove it if you have installed GNOME Doc Utils.

`--enable-xhost-root`: This switch provides an interim workaround to allow GParted to run under Wayland by using xhost to grant and revoke root access to the X11 display.

Using Gparted

To manipulate file systems Gparted has a run time dependency on various file system tools (you only need to install the tools for file systems you actually use): [Hdparm-9.65](#) (required for optionally displaying serial number device information), [btrfs-progs-6.10.1](#), [dosfstools-4.2](#), e2fsprogs (installed as part of LFS), [jfsutils-1.1.15](#), [xfsprogs-6.9.0](#), [mtools](#) (required to read and write FAT16/32 volume labels and UUIDs), [hfsutils](#), [hfsprogs](#), [nilfs-utils](#), [Reiser4progs](#), and [reiserfsprogs](#).

Root privileges are required to run Gparted. If you wish to run the application from the menu, further applications and configurations are necessary. Examples of applications that may be used: [gksu](#), [kdesudo](#), or [xdg-su](#). Another simple solution is [ssh-askpass-9.8p1](#).

ssh-askpass

To optionally use [ssh-askpass-9.8p1](#) if it is installed in your system, run the following commands as the `root` user:

```
cp -v /usr/share/applications/gparted.desktop /usr/share/applications/gparted.desktop.back &&
sed -i 's/Exec=/Exec=sudo -A /' /usr/share/applications/gparted.desktop
```

Now, clicking on the menu item for Gparted, a dialog appears on the screen, asking for the administrator password.

Contents

Installed Programs: gparted and gparted_polkit (optional)

Installed Libraries: None

Installed Directories: None

Short Descriptions

`gparted` is a shell script which sets up the environment before calling `gpartedbin`
`gparted_polkit` is an optional script which can be used to run gparted with polkit, from a menu

Inkscape-1.3.2

Introduction to Inkscape

Inkscape is a what you see is what you get Scalable Vector Graphics editor. It is useful for creating, viewing and changing SVG images.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://inkscape.org/gallery/item/44615/inkscape-1.3.2.tar.xz>
- Download MD5 sum: 76ed1f4b13065e80de8b2d77b6427b83
- Download size: 43 MB
- Estimated disk space required: 817 MB (189 MB installed)
- Estimated build time: 4.6 SBU (with parallelism=8)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/inkscape-1.3.2-upstream_fixes-2.patch

Note

The tarball `inkscape-1.3.2.tar.xz` will extract to the directory `inkscape-1.3.2_2023-11-25_091e20ef0f/`.

Inkscape Dependencies

Required

[Boost-1.86.0](#), [double-conversion-3.3.0](#), [GC-8.2.6](#), [gsl-2.8](#), [Gtkmm-3.24.9](#), [libsoup-2.74.3](#), [libxslt-1.1.42](#), [Poppler-24.08.0](#), and [popt-1.19](#)

Recommended

[ImageMagick-7.1.1-36](#) (runtime), [Little CMS-2.16](#), [libcanberra-0.30](#) (to eliminate plugin warnings), [Potrace-1.16](#) (for the bucket-fill tool), also various Python modules at runtime for the core extensions: [CacheControl-0.14.0](#), [cssselect-1.2.0](#), [lxml-5.3.0](#), [NumPy-2.1.0](#), [pySerial-3.5](#), and [Scour-0.38.2](#)

Optional

[Aspell-0.60.8.1](#), [dbus-1.14.10](#) (to run inkscape from scripts), [Doxygen-1.12.0](#), [gspell-1.12.2](#), [GraphicsMagick](#), [libcdr](#), [libvisio](#), [libwpg](#) (or [libwpd](#))

Installation of Inkscape

First, fix Inkscape to build with poppler-24.05.0:

```
patch -Np1 -i ../../inkscape-1.3.2-upstream_fixes-2.patch
```

Next, fix a build failure caused by libxml2-2.12.0:

```
sed -i '/uri.h/a #include <libxml/xmlmemory.h>' src/object/uri.h
```

Install Inkscape by running the following commands:

```
mkdir build &&
cd build &&
```

```
cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      ..
      &&
make
```

Note

This package may occasionally fail when building with multiple processors. See [Using Multiple Processors](#) for more information.

This package does not come with a working test suite.

Now, as the `root` user:

```
make install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+-3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

`-D CMAKE_BUILD_TYPE=Release`: This switch is used to build the release library without any debug `assert` in the code.

`-D WITH_DBUS=ON`: use this if you wish to use `inkscape` in interactive scripts which manipulate images.

Contents

Installed Programs: `inkscape` and `inkview`

Installed Library: `libinkscape_base.so` (in `/usr/lib/inkscape`)

Installed Directories: `/usr/lib/inkscape`, and `/usr/share/inkscape`

Short Descriptions

<code>inkscape</code>	is a SVG (Scalable Vector Graphics) editing program
<code>inkview</code>	is a simple program for displaying SVG files
<code>libinkscape_base.so</code>	provides the routines used by <code>inkscape</code> and <code>inkview</code>

rxvt-unicode-9.31

Introduction to rxvt-unicode

`rxvt-unicode` is a clone of the terminal emulator `rxvt`, an X Window System terminal emulator which includes support for XFT and Unicode.

This package is known to build and work properly using an LFS 12.2 platform.

Note

Please be aware that `rxvt-unicode` is affected by an issue in perl where the `SIGFPE` handler is set to `SIG_IGN` (i.e. the signal is ignored). If you are building an LFS system from within `urxvt` and running the test suites, tests in bash and check which test raising this signal will fail.

Package Information

- Download (HTTP): <http://dist.schmorp.de/rxvt-unicode/Attic/rxvt-unicode-9.31.tar.bz2>
- Download MD5 sum: 3d0ec83705c9b9ff301a4b9965b3cd9f
- Download size: 860 KB
- Estimated disk space required: 22 MB
- Estimated build time: 0.2 SBUs

rxvt-unicode Dependencies

Required

[libptytty-2.0](#) and [a graphical environment](#)

Optional

[gdk-pixbuf-2.42.12](#) (for background images) and [startup-notification-0.12](#)

Installation of rxvt-unicode

Install rxvt-unicode by running the following commands:

```
./configure --prefix=/usr --enable-everything &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-everything`: Add support for all non-multichoice options. Details about the different options can be found in the file `README.configure`.

`--disable-xft`: Remove support for Xft fonts.

`--disable-perl`: Disable the embedded Perl interpreter.

`--disable-afterimage`: Remove support for libAfterImage.

Configuring rxvt-unicode

The rxvt-unicode terminal emulator uses the resource class `URxvt` and the resource name `urxvt`. You can add X resource definitions to a user's `~/.Xresources` or `~/.Xdefaults` files or to the system-wide `/etc/X11/app-defaults/URxvt` file. The following example will load the `matcher` Perl extension (assuming Perl support wasn't disabled), which enables a left button click to open an underlined URL in the specified browser, sets a background and foreground color and loads an Xft font (as the `root` user):

```
cat >> /etc/X11/app-defaults/URxvt << "EOF"
! Use the specified colour as the windows background colour [default white]
URxvt*background: black

! Use the specified colour as the windows foreground colour [default black]
URxvt*foreground: yellow

! Select the fonts to be used. This is a comma separated list of font names
URxvt*font: xft:Monospace:pixelsize=18

! Comma-separated list(s) of perl extension scripts (default: "default")
URxvt*perl-ext: matcher

! Specifies the program to be started with a URL argument. Used by
URxvt?url-launcher: firefox

! When clicked with the mouse button specified in the "matcher.button" resource
! (default 2, or middle), the program specified in the "matcher.launcher"
! resource (default, the "url-launcher" resource, "sensible-browser") will be
! started with the matched text as first argument.
```

```
! Below, default modified to mouse left button.  
URxvt*matcher.button:      1  
EOF
```

In order to view the defined X resources, issue:

```
xrdb -query
```

In order to add the modifications of the new user configuration file, keeping previously X configurations (of course, unless you are changing any on previously define), issue:

```
xrdb -merge ~/.Xresources
```

The rxvt-unicode application can also run in a daemon mode, which makes it possible to open multiple terminal windows within the same process. The `urxvtd` client then connects to the `urxvtd` daemon and requests a new terminal window. Use this option with caution. If the daemon crashes, all the running processes in the terminal windows are terminated.

You can start the `urxvtd` daemon in the system or personal startup x session script (e.g., `~/.xinitrc`) by adding the following lines near the top of the script:

```
# Start the urxvtd daemon  
urxvtd -q -f -o &
```

For more information, examine the `urxvt`, `urxvtd`, `urxvtc`, and `urxvtperl` man pages.

If you use a Desktop Environment, a menu entry can be included, issuing, as the `root` user:

```
cat > /usr/share/applications/urxvt.desktop << "EOF" &&  
[Desktop Entry]  
Encoding=UTF-8  
Name=Rxvt-Unicode Terminal  
Comment=Use the command line  
GenericName=Terminal  
Exec=urxvt  
Terminal=false  
Type=Application  
Icon=utilities-terminal  
Categories=GTK;Utility;TerminalEmulator;  
#StartupNotify=true  
Keywords=console;command line;execute;  
EOF  
  
update-desktop-database -q
```

For that, you need [desktop-file-utils-0.27](#) and at least one of [gnome-icon-theme-3.12.0](#), [oxygen-icons-6.0.0](#) and [lxde-icon-theme-0.5.1](#). Uncomment the line with "StartupNotify=true" if [startup-notification-0.12](#) is installed.

Contents

Installed Programs: urxvt, urxvtd, and urxvtc

Installed Libraries: Many Perl extensions located under `/usr/lib/urxvt/perl`

Installed Directory: `/usr/lib/urxvt`

Short Descriptions

`urxvt` is a terminal emulator for the X Window System
`urxvtd` is the `urxvt` terminal daemon
`urxvtc` controls the `urxvtd` daemon

Thunderbird-128.1.0esr

Introduction to Thunderbird

Thunderbird is a stand-alone mail/news client based on the Mozilla codebase. It uses the Gecko rendering engine to enable it to display and compose HTML emails.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.mozilla.org/pub/thunderbird/releases/128.1.0esr/source/thunderbird-128.1.0esr.source.tar.xz>
- Download MD5 sum: 77dedc2267784055d1bdc27916660177
- Download size: 646 MB
- Estimated disk space required: 7.8 GB (256 MB installed)
- Estimated build time: 16 SBU (on a 8-core machine)

Thunderbird Dependencies

Required

[Cbindgen-0.27.0](#), [GTK+-3.24.43](#), [LLVM-18.1.7](#) including clang, [nodejs-20.16.0](#), [PulseAudio-17.0](#) (or [alsa-lib-1.2.12](#) if you edit the mozconfig; although it is now deprecated by mozilla), [Python-3.12.5](#) (rebuilt with the sqlite module), [startup-notification-0.12](#), and [UnZip-6.0](#)

Recommended

[libevent-2.1.12](#), [libvpx-1.14.1](#), [libwebp-1.4.0](#), [NASM-2.16.03](#), [NSPR-4.35](#), and [nss-3.103](#)

Note

If you don't install recommended dependencies, then internal copies of those packages will be used. They might be tested to work, but they can be out of date or contain security holes.

Optional

[pciutils-3.13.0](#) (runtime), [Wget-1.24.5](#), [Wireless Tools-29](#), and [watchman](#)

Installation of Thunderbird

Note

The build process for Thunderbird can use 8 GB or more of RAM when linking. Make sure that you have adequate swap or RAM before continuing.

The configuration of Thunderbird is accomplished by creating a `mozconfig` file containing the desired configuration options. A default `mozconfig` is created below. To see the entire list of available configuration options (and a brief description of each), issue `./mach configure -- --help | less`. Create the file with the following command:

```
cat > mozconfig << "EOF"
# If you have a multicore machine, all cores will be used.

# If you have installed wireless-tools comment out this line:
ac_add_options --disable-necko-wifi

# Uncomment the following option if you have not installed PulseAudio
#ac_add_options --enable-audio-backends=alsa

# Comment out following options if you have not installed
# recommended dependencies:
ac_add_options --with-system-libevent
ac_add_options --with-system-libvpx
ac_add_options --with-system-nspr
ac_add_options --with-system-nss
ac_add_options --with-system-webp

# The BLFS editors recommend not changing anything below this line:
ac_add_options --prefix=/usr
ac_add_options --enable-application=comm/mail

ac_add_options --disable-crashreporter
ac_add_options --disable-updater
ac_add_options --disable-debug
```

```

ac_add_options --disable-debug-symbols
ac_add_options --disable-tests

# This enables SIMD optimization in the shipped encoding_rs crate.
ac_add_options --enable-rust-simd

ac_add_options --enable-strip
ac_add_options --enable-install-strip

# You cannot distribute the binary if you do this.
ac_add_options --enable-official-branding

ac_add_options --enable-system-ffi
ac_add_options --enable-system-pixman

ac_add_options --with-system-jpeg
ac_add_options --with-system-png
ac_add_options --with-system-zlib

# Using sandboxed wasm libraries has been moved to all builds instead
# of only mozilla automation builds. It requires extra llvm packages
# and was reported to seriously slow the build. Disable it.
ac_add_options --without-wasm-sandboxed-libraries
EOF

```

Adapt two files for compatibility with cbindgen-0.27.0 or later:

```

sed 's/input.try/!& parse/' \
-i servo/components/style_traits/values.rs &&
sed '0,/"/Keyframe"/{ //d}' -i servo/ports/geckolib/cbindgen.toml

```

Now invoke the Python `mach` script to compile Thunderbird:

Note

If you are compiling this package in chroot you must ensure that `/dev/shm` is mounted. If you do not do this, the Python configuration will fail with a traceback report referencing `/usr/lib/pythonN.N/multiprocessing/synchronize.py`. As the `root` user, run:

```
mountpoint -q /dev/shm || mount -t tmpfs devshm /dev/shm
```

```

export MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE=none &&
export MOZBUILD_STATE_PATH=$(pwd)/mozbuild &&
./mach build

```

This package does not come with a test suite.

Install Thunderbird by running the following commands as the `root` user:

```
MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE=none ./mach install
```

Empty the environment variables which were set above:

```

unset MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE &&
unset MOZBUILD_STATE_PATH

```

Command Explanations

`MACH_BUILD_PYTHON_NATIVE_PACKAGE_SOURCE=none`: Use the system python to create a virtual environment for `mach` without downloading any python wheels nor using the system python modules. This prevent version mismatches between system modules and bundled ones.

`./mach build --verbose`: Use this alternative if you need details of which files are being compiled, together with any C or C++ flags being used.

`./mach build -jN`: The build should, by default, use all the online CPU cores. If using all the cores causes the build to swap because you have insufficient memory, using fewer cores can be faster.

`CC=gcc CXX=g++`: BLFS used to prefer to use gcc and g++ instead of upstream's defaults of the clang programs. With the release of gcc-12 the build takes longer with gcc and g++, primarily because of extra warnings, and is bigger. Set these

environment variables before you run the configure script if you wish to continue to use gcc, g++. Building with GCC on i? 86 is currently broken.

Configuring Thunderbird

Configuration Information

If your Window Manager or Desktop Environment does not allow you to configure a default browser, you can add a configuration parameter to Thunderbird so that a browser will start when you click on an Internet/intranet/local URL. The procedure to check or modify any of the configuration parameters is quite simple and the instructions here can be used to view or modify any of the parameters.

First, open the configuration dialog by opening the "Edit" drop-down menu. Choose "Settings" and then scroll down to the bottom of the page. Then, click the "Config Editor" button. Click on the "I accept the risk!" button. This will display a list of the configuration preferences and information related to each one. You can use the "Filter:" bar to enter search criteria and narrow down the listed items. Changing a preference can be done using two methods. One, if the preference has a boolean value (True/False), simply double-click on the preference to toggle the value and two, for other preferences simply right-click on the desired line, choose "Modify" from the menu and change the value. Creating new preference items is accomplished in the same way, except choose "New" from the menu and provide the desired data into the fields when prompted.

If you use a desktop environment such as GNOME or KDE, a desktop file `thunderbird.desktop` may be created, in order to include a "Thunderbird" entry in the menu. Run the following commands as the `root` user:

```
mkdir -pv /usr/share/{applications,pixmaps} &&
cat > /usr/share/applications/thunderbird.desktop << "EOF" &&
[Desktop Entry]
Name=Thunderbird Mail
Comment=Send and receive mail with Thunderbird
GenericName=Mail Client
Exec=thunderbird %u
Terminal=false
Type=Application
Icon=thunderbird
Categories=Network;Email;
MimeType=text/html;text/xml;application/xhtml+xml;application/xml;application/rss+xml;x-scheme-handler/mailto;
StartupNotify=true
EOF
ln -sfv /usr/lib/thunderbird/chrome/icons/default/default256.png \
/usr/share/pixmaps/thunderbird.png
```

Contents

Installed Program: `thunderbird`

Installed Libraries: Numerous libraries and modules in the `/usr/lib/thunderbird` directory

Installed Directory: `/usr/lib/thunderbird`

Short Descriptions

`thunderbird` is Mozilla's email and newsgroup client

Tigervnc-1.14.0

Introduction to Tigervnc

Tigervnc is an advanced VNC (Virtual Network Computing) implementation. It allows creation of an Xorg server not tied to a physical console and also provides a client for viewing of the remote graphical desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/TigerVNC/tigervnc/archive/v1.14.0/tigervnc-1.14.0.tar.gz>
- Download MD5 sum: 402c2ec1f38bd0fb4ad7d93631dc1502
- Download size: 2.0 MB
- Estimated disk space required: 101 MB

- Estimated build time: 0.5 SBU (Using parallelism=4)

Additional Downloads

- Required file: <https://www.x.org/pub/individual/xserver/xorg-server-21.1.13.tar.xz>

Note

The version of Xorg may eventually be a few versions out of date, but is required for the customizations needed for this package.

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/tigervnc-1.14.0-configuration_fixes-1.patch
- Optional file to start the server without a display manager:
<https://anduin.linuxfromscratch.org/BLFS/tigervnc/Xsession>

Tigervnc Dependencies

Required

[CMake-3.30.2](#), [FLTK-1.3.9](#), [GnuTLS-3.8.7.1](#), [libgcrypt-1.11.0](#), [libjpeg-turbo-3.0.1](#), [Pixman-0.43.4](#), [Systemd-256.4](#) (with [Linux-PAM-1.6.1](#)), [Xorg Applications](#), [xinit-1.4.2](#), and [Xorg Legacy Fonts](#)

Recommended

[ImageMagick-7.1.1-36](#)

Installation of Tigervnc

First, make adjustments to the configuration files to make them compatible with LFS systems:

```
patch -Npl -i ../../tigervnc-1.14.0-configuration_fixes-1.patch
```

Install tigervnc by running the following commands:

```
# Put code in place
mkdir -p unix/xserver &&
tar -xf ../../xorg-server-21.1.13.tar.xz \
    --strip-components=1 \
    -C unix/xserver &&
( cd unix/xserver &&
  patch -Npl -i ../../xserver21.1.1.patch ) &&

# Build viewer
cmake -G "Unix Makefiles" \
      -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -W no-dev . &&
make &&

# Build server
pushd unix/xserver &&
  autoreconf -fiv &&

  CPPFLAGS="-I/usr/include/drm" \
  ./configure $XORG_CONFIG \
    --disable-xwayland --disable-dri --disable-dmx \
    --disable-xorg --disable-xnest --disable-xvfb \
    --disable-xwin --disable-xephyr --disable-kdrive \
    --disable-devel-docs --disable-config-hal --disable-config-udev \
    --disable-unit-tests --disable-selective-werror \
    --disable-static --enable-dri3 \
    --without-dtrace --enable-dri2 --enable-glx \
    --with-pic &&
  make &&
popd
```

This package does not come with a test suite.

Now, as the `root` user:

```
#Install viewer  
make install &&  
mv /usr/share/doc/tigervnc /usr/share/doc/tigervnc-1.14.0  
  
#Install server  
( cd unix/xserver/hw/vnc && make install ) &&  
  
[ -e /usr/bin/Xvnc ] || ln -svf $XORG_PREFIX/bin/Xvnc /usr/bin/Xvnc
```

Command Explanations

`tar -xf ... xorg-server...:` This command extracts the standard Xorg package into the tree in a location needed for modification.

`--disable ...:` Most options that are usually needed for the standard Xorg server are not needed for the Xvnc instance being built.

`[-e /usr/bin/Xvnc] || ln ... xvnc:` If the Xvnc server is not installed in the `/usr/bin` directory, then create a link so the `vncserver` script can find it.

Configuring Tigervnc

Server Configuration

On systemd systems, another method of configuration is available. This configuration provides the added benefit of making `tigervnc` systemd aware for VNC sessions and allows desktop environments like GNOME to autostart services once the VNC session is started. This configuration also gives the added benefit of starting VNC Sessions on system startup. To set up the VNC server in this fashion, follow these instructions.

First, install a rudimentary Xsession file so that the VNC server can initialize X sessions properly:

```
install -vdm755 /etc/X11/tigervnc &&  
install -v -m755 ..../Xsession /etc/X11/tigervnc
```

Next, set up a user mapping in `/etc/tigervnc/vncserver.users`. This tells the VNC Server which session is allocated to a user.

```
echo ":1=$(whoami)" >> /etc/tigervnc/vncserver.users
```

Next, set up a configuration file to tell `vncserver` which desktop environment should be used and what display geometry should be used. There are several other options that can be defined in this file, but they are outside the scope of BLFS.

```
install -vdm 755 ~/.vnc &&  
cat > ~/.vnc/config << EOF  
# Begin ~/.vnc/config  
# The session must match one listed in /usr/share/xsessions.  
# Ensure that there are no spaces at the end of the lines.  
  
session=lxqt  
geometry=1024x768  
  
# End ~/.vnc/config  
EOF
```

To start the VNC Server, run the following command:

```
systemctl start vncserver@:1
```

To start the VNC Server when the system boots, run the following command:

```
systemctl enable vncserver@:1
```

Contents

Installed Programs: Xvnc, vncconfig, vncpasswd, vncserver, vncviewer, and x0vncserver

Installed Libraries: libvnc.so

Installed Directories: /usr/share/doc/tigervnc-1.14.0

Short Descriptions

xvnc	is a X VNC (Virtual Network Computing) server. It is based on a standard X server, but it has a "virtual" screen rather than a physical one
vncconfig	is a program to configure and control a VNC server
vncpasswd	allows you to set the password used to access VNC desktops
vncserver	is a perl script used to start or stop a VNC server
vncviewer	is a client used to connect to VNC desktops
x0vncserver	is a program to make an X display on a physical terminal accessible via TigerVNC or compatible viewers

Transmission-4.0.6

Introduction to Transmission

Transmission is a cross-platform, open source BitTorrent client. This is useful for downloading large files (such as Linux ISOs) and reduces the need for the distributors to provide server bandwidth.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/transmission/transmission/releases/download/4.0.6/transmission-4.0.6.tar.xz>
- Download MD5 sum: 8132b9f012b8e6309911c80ee9fd00f7
- Download size: 11 MB
- Estimated disk space required: 226 MB (with both GUI interfaces)
- Estimated build time: 2.3 SBU (with both GUI interfaces; using parallelism=4)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/transmission-4.0.6-build_fix-1.patch

Transmission Dependencies

Required

[cURL-8.9.1](#)

Recommended

[libevent-2.1.12](#) and [libpsl-0.21.5](#) (to use system libraries instead of bundled ones)

Recommended (to build a GUI)

[Gtkmm-4.14.0](#) or [Qt-6.7.2](#)

Optional

[nodejs-20.16.0](#) (for building the web client, not needed at run time), [appindicator](#), [dht](#), [libb64](#), [libdeflate](#), [libnatpmp](#), [libutp](#), and [miniuupnp](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/transmission>

Installation of Transmission

First, fix a build failure due to incorrect statements in CMakeLists.txt:

```
patch -Np1 -i ../../transmission-4.0.6-build_fix-1.patch
```

Install Transmission by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
```

```
-D CMAKE_INSTALL_DOCDIR=/usr/share/doc/transmission-4.0.6 \
.. &&
make
```

Now, as the `root` user:

```
make install
```

Next, create `transmission.png` from the SVG file:

```
rsvg-convert \
/usr/share/icons/hicolor/scalable/apps/transmission.svg \
-o /usr/share/pixmaps/transmission.png
```

Command Explanations

`-D ENABLE_QT=OFF`: This switch disables building the Qt interface. The default is to build it if [qt5-components-5.15.14](#) or [Qt-6.7.2](#) is installed.

`-D ENABLE_GTK=OFF`: This switch disables building the GTK-4 interface. The default is to build it if [Gtkmm-4.14.0](#) is installed.

`-D ENABLE_WEB=OFF`: This switch disables building the web client. The default is to build it if [nodejs-20.16.0](#) is installed.

`-D REBUILD_WEB=ON`: This switch forces rebuilding the web client. This option needs [nodejs-20.16.0](#) to be installed as well as an internet connection. The default is to not rebuild the client.

Contents

Installed Programs: `transmission-create`, `transmission-daemon`, `transmission-edit`, `transmission-gtk`, `transmission-qt`, `transmission-remote`, and `transmission-show`

Installed Libraries: None

Installed Directory: `/usr/share/transmission` (contains the web client) and `/usr/share/doc/transmission-4.0.6`

Short Descriptions

<code>transmission-create</code>	is a command line tool used to create .torrent files
<code>transmission-daemon</code>	is a daemon-based Transmission session that can be controlled via RPC commands from Transmission's web interface or <code>transmission-remote</code>
<code>transmission-edit</code>	is a command line tool to modify .torrent files' announce URLs
<code>transmission-gtk</code>	is a GTK+ bittorrent client
<code>transmission-qt</code>	is a Qt-based bittorrent client
<code>transmission-remote</code>	is a remote control utility for transmission-daemon and transmission
<code>transmission-show</code>	is a command line tool to display bittorrent .torrent file metadata

xarchiver-0.5.4.23

Introduction to xarchiver

XArchiver is a GTK+ archive manager with support for tar, xz, bzip2, gzip, zip, 7z, rar, lzo and many other archive formats.

This package is known to build and work properly using an LFS 12.2 platform.

Note

xarchiver is only a command line or graphical interface to archiving utilities such as tar and zip.

Package Information

- Download (HTTP): <https://github.com/ib/xarchiver/archive/0.5.4.23/xarchiver-0.5.4.23.tar.gz>

- Download MD5 sum: bfdace2b113417e1bc77ef304280d103
- Download size: 1.1 MB
- Estimated disk space required: 11 MB
- Estimated build time: 0.1 SBU

xarchiver Dependencies

Required

[GTK+-3.24.43](#)

Optional (Runtime)

[cpio-2.15](#), [LZO-2.10](#), [p7zip-17.04](#), [UnRar-7.0.9](#), [UnZip-6.0](#), and [Zip-3.0](#)

Installation of xarchiver

Install xarchiver by running the following commands:

```
./configure --prefix=/usr \
--libexecdir=/usr/lib/xfce4 \
--docdir=/usr/share/doc/xarchiver-0.5.4.23 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+-3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

`--libexecdir=/usr/lib/xfce4`: This switch fixes the location for the thunar-archive-plugin, so it will work if [thunar-4.18.11](#) is installed.

Contents

Installed Program: xarchiver

Installed Libraries: None

Installed Directories: /usr/lib/xfce4/thunar-archive-plugin, /usr/share/doc/xarchiver-0.5.4.23, and /usr/share/pixmaps/xarchiver

Short Descriptions

`xarchiver` is a GTK+ archive manager

xdg-utils-1.2.1

Introduction to xdg-utils

`xdg-utils` is a set of command line tools that assist applications with a variety of desktop integration tasks. It is required for Linux Standards Base (LSB) conformance.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/xdg/xdg-utils/-/archive/v1.2.1/xdg-utils-v1.2.1.tar.gz>
- Download MD5 sum: 4c72585a98ba8f775cb9e72b066cc0df
- Download size: 304 KB
- Estimated disk space required: 3.3 MB (with tests)
- Estimated build time: 2.5 SBU (with tests)

xdg-utils Dependencies

Required

[xmlto-0.0.29](#) with one of [Lynx-2.9.2](#), [Links-2.30](#), or [W3m](#)

Required (runtime)

[Xorg Applications](#)

Optional (runtime)

[dbus-1.14.10](#)

Installation of `xdg-utils`

Compile `xdg-utils` with the following commands:

```
./configure --prefix=/usr &&  
make
```

Caution

The tests for the scripts must be made from an X-Window based session. There are several run-time requirements to run the tests including a browser and an MTA. Running the tests as `root` user is not recommended.

To run the tests, issue: `make -k test`.

Now install it as the `root` user:

```
make install
```

Contents

Installed Programs: `xdg-desktop-menu`, `xdg-desktop-icon`, `xdg-mime`, `xdg-icon-resource`, `xdg-open`, `xdg-email`, `xdg-screensaver`, `xdg-settings`

Installed Libraries: None

Installed Directories: None

Short Descriptions

<code>xdg-desktop-menu</code>	is a command line tool for (un)installing desktop menu items
<code>xdg-desktop-icon</code>	is a command line tool for (un)installing icons to the desktop
<code>xdg-mime</code>	is a command line tool for querying information about file type handling and adding descriptions for new file types
<code>xdg-icon-resource</code>	is a command line tool for (un)installing icon resources

<code>xdg-open</code>	opens a file or URL in the user's preferred application
<code>xdg-email</code>	opens the user's preferred e-mail composer in order to send a mail message
<code>xdg-screensaver</code>	is a command line tool for controlling the screensaver
<code>xdg-settings</code>	is a command line tool for managing various settings from the desktop environment

XScreenSaver-6.09

Introduction to XScreenSaver

The XScreenSaver package is a modular screen saver and locker for the X Window System. It is highly customizable and allows the use of any program that can draw on the root window as a display mode. The purpose of XScreenSaver is to display pretty pictures on your screen when it is not in use, in keeping with the philosophy that unattended monitors should always be doing something interesting, just like they do in the movies. However, XScreenSaver can also be used as a screen locker, to prevent others from using your terminal while you are away.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.jwz.org/xscreensaver/xscreensaver-6.09.tar.gz>
- Download MD5 sum: 1f0e2b1698bc2a6ecfe9b668e4b75c3d
- Download size: 21 MB
- Estimated disk space required: 262 MB
- Estimated build time: 0.4 SBU (Using parallelism=4)

XScreenSaver Dependencies

Required

[GTK+-3.24.43](#) and [Xorg Applications](#)

Recommended

[GLU-9.0.3](#)

Optional

[GDM-46.2](#), [FFmpeg-7.0.2](#), [Linux-PAM-1.6.1](#), [MIT Kerberos V5-1.21.3](#), and [GLE](#)

Installation of XScreenSaver

Install XScreenSaver by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

--with-setuid-hacks: This switch allows some demos to be installed setuid `root` which is needed in order to ping other hosts.

Configuring XScreenSaver

Config Files

`/etc/X11/app-defaults/XScreenSaver` and `~/.xscreensaver`

Linux PAM Configuration

If XScreenSaver has been built with Linux PAM support, you need to create a PAM configuration file, to get it working correctly with BLFS.

Issue the following commands as the `root` user to create the configuration file for Linux PAM:

```
cat > /etc/pam.d/xscreensaver << "EOF"
# Begin /etc/pam.d/xscreensaver

auth    include system-auth
account include system-account

# End /etc/pam.d/xscreensaver
EOF
```

Contents

Installed Programs: `xscreensaver`, `xscreensaver-command`, `xscreensaver-demo`, and `xscreensaver-settings`

Installed Libraries: None

Installed Directories: `/usr/libexec/xscreensaver` and `/usr/share/xscreensaver`

Short Descriptions

<code>xscreensaver</code>	is a screen saver and locker daemon
<code>xscreensaver-command</code>	controls a running <code>xscreensaver</code> process by sending it client messages
<code>xscreensaver-demo</code>	is a symlink to <code>xscreensaver-settings</code>
<code>xscreensaver-settings</code>	is a graphical front-end for setting the parameters used by the background <code>xscreensaver</code> daemon

Part XII. Multimedia

Chapter 42. Multimedia Libraries and Drivers

Many multimedia programs require libraries and/or drivers in order to function properly. The packages in this section fall into this category. Generally you only need to install these if you are installing a program which has the library listed as either a requirement, or as an option to enable it to support certain functionality.

ALSA

The Linux kernel now provides ALSA support by default. However, applications need to interface to that capability. The following six sections of the book deal with the separate components of ALSA: the libraries, the plugins, the utilities, the tools, and the firmware.

alsa-lib-1.2.12

Introduction to ALSA Library

The ALSA Library package contains the ALSA library used by programs (including ALSA Utilities) requiring access to the ALSA sound interface.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.alsa-project.org/files/pub/lib/alsa-lib-1.2.12.tar.bz2>
- Download MD5 sum: 97bbf3cca911c1701fe9fe115c32b03d
- Download size: 1.1 MB
- Estimated disk space required: 46 MB (with tests and docs)
- Estimated build time: 0.2 SBU (with tests and docs)

Additional Downloads

- Recommended file: <https://www.alsa-project.org/files/pub/lib/alsa-ucm-conf-1.2.12.tar.bz2>

ALSA Library Dependencies

Recommended (Runtime)

[Systemd-256.4](#)

Note

If the recommended runtime dependency is not installed, you may need to run any application requiring ALSA library as the `root` user or a user in the `audio` group.

Optional

[Doxygen-1.12.0](#) and [Python2](#)

Kernel Configuration

If needed, enable the following options in the kernel configuration and recompile the kernel:

```
Device Drivers --->
  <*/M> Sound card support --->                                [SOUND]
    # Select settings and drivers appropriate for your hardware
    # in the submenu:
  <*/M> Advanced Linux Sound Architecture --->                [SND]
```

In the Device Drivers ⇒ Sound card support ⇒ Advanced Linux Sound Architecture section of the kernel configuration, select the settings and drivers appropriate for your hardware. If necessary, recompile and install your new kernel.

Installation of ALSA Library

Install ALSA Library by running the following commands:

```
./configure &&
make
```

If you have Doxygen installed and you wish to build the library API documentation, run the following commands from the top-level directory of the source tree:

```
make doc
```

To test the results, issue: `make check`.

Now, as the `root` user, install the package and recommended configuration files:

```
make install &&
tar -C /usr/share/alsa --strip-components=1 -xf ../alsa-ucm-conf-1.2.12.tar.bz2
```

To install the API documentation, run the following command as the `root` user:

```
install -v -d -m755 /usr/share/doc/alsa-lib-1.2.12/html/search &&
install -v -m644 doc/doxygen/html/*.* \
          /usr/share/doc/alsa-lib-1.2.12/html &&
install -v -m644 doc/doxygen/html/search/* \
          /usr/share/doc/alsa-lib-1.2.12/html/search
```

Configuring ALSA Library

Config Files

`~/.asoundrc` and `/etc/asound.conf`

Configuration Information

The default `alsa.conf` is adequate for most installations. For extra functionality and/or advanced control of your sound device, you may need to create additional configuration files. For information on the available configuration parameters, visit <https://www.alsa-project.org/main/index.php/Asoundrc>.

Contents

Installed Program: `aserver`

Installed Library: `libasound.so` and `libatopology.so`

Installed Directories: `/usr/include/alsa`, `/usr/share/alsa` and `/usr/share/doc/alsa-lib-1.2.12`

Short Descriptions

<code>aserver</code>	is the ALSA server
<code>libasound.so</code>	contains the ALSA API functions
<code>libatopology.so</code>	contains API functions for the ALSA topologies

alsa-plugins-1.2.12

Introduction to ALSA Plugins

The ALSA Plugins package contains plugins for various audio libraries and sound servers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.alsa-project.org/files/pub/plugins/alsa-plugins-1.2.12.tar.bz2>
- Download MD5 sum: 71fce75baa38efab4e32e40db0a33716
- Download size: 400 KB
- Estimated disk space required: 5.1 MB
- Estimated build time: less than 0.1 SBU

ALSA Plugins Dependencies

Required

[alsa-lib-1.2.12](#)

Optional

[FFmpeg-7.0.2](#), [libsamplerate-0.2.2](#), [PulseAudio-17.0](#), [Speex-1.2.1](#), [JACK](#), [libavtp](#), and [maemo](#)

Installation of ALSA Plugins

Install ALSA Plugins by running the following commands:

```
./configure --sysconfdir=/etc &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: Numerous libasound_module_<module>.so modules including conf_pulse, ctl_arcam_av, ctl_oss, ctl_pulse, pcm_a52, pcm_jack, pcm_oss, pcm_pulse, pcm_speex, pcm_upmix, pcm_usb_stream, pcm_vdownmix, rate_lavrare*, rate_samplerate*, and rate_speexrate*

Installed Directories: `/usr/lib/alsa-lib`

Short Descriptions

<code>libasound_module_pcm_oss.so</code>	Allows native ALSA applications to run on OSS
<code>libasound_module_pcm_upmix.so</code>	Allows upmixing sound to 4 or 6 channels
<code>libasound_module_pcm_vdownmix.so</code>	Allows downmixing sound from 4-6 channels to 2 channel stereo output
<code>libasound_module_pcm_jack.so</code>	Allows native ALSA applications to work with <code>jackd</code>
<code>libasound_module_pcm_pulse.so</code>	Allows native ALSA applications to access a PulseAudio sound daemon
<code>libasound_module_pcm_a52.so</code>	Converts S16 linear sound format to A52 compressed format and sends it to an SPDIF output
<code>libasound_module_rate_samplerate.so</code>	Provides an external rate converter through <code>libsamplerate</code>

alsa-utils-1.2.12

Introduction to ALSA Utilities

The ALSA Utilities package contains various utilities which are useful for controlling your sound card.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.alsa-project.org/files/pub/utils/alsa-utils-1.2.12.tar.bz2>
- Download MD5 sum: 80b71081f0bd2bb688e2f04311f142c5
- Download size: 1.6 MB
- Estimated disk space required: 16 MB
- Estimated build time: 0.3 SBU

ALSA Utilities Dependencies

Required

[alsa-lib-1.2.12](#)

Optional

[docutils-0.21.2](#), [fftw-3.3.10](#), [libsamplerate-0.2.2](#), [xmlto-0.0.29](#), and [Dialog](#)

Installation of ALSA Utilities

Install ALSA Utilities by running the following commands:

```
./configure --disable-alsaconf \
--disable-bat \
--disable-xmlto \
--with-curses=ncursesw &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-alsaconf`: This switch disables building the `alsaconf` configuration tool which is incompatible with Udev.

`--disable-xmlto`: Omit this switch if you have installed [xmlto-0.0.29](#) and wish to regenerate the man pages.

`--disable-bat`: Omit this switch if you have installed [fftw-3.3.10](#) and wish to install the Basic Audio Tester (BAT).

`--with-curses=ncursesw`: This switch forces the use of wide-character ncurses libraries.

Configuring ALSA Utilities

Config Files

```
/var/lib/alsa/asound.state
```

Configuration Information

As the `root` user, apply the default setting for the sound card. Note that this command may return an exit code of 99 although the card is correctly initialized:

```
alsactl init
```

Note that all channels of your sound card may be muted by default. You can use the `alsamixer` program to change this. Use `speaker-test` to check that your settings have been applied correctly. You should hear "pink noise" on your speakers.

The `alsactl` program is normally run from a standard udev rule. The first time it is run, it will complain that there is no state in `/var/lib/alsa/asound.state`. You can prevent this by running the following command as the `root` user:

```
alsactl -L store
```

The volume settings should be restored from the saved state by Udev when the device is detected (during boot or when plugged in for USB devices).

On systems which have multiple sound cards, you may need to adjust the default audio device so that you can get output from your speakers. To set the default device, first check the `/proc/asound/cards` file to determine which number you need to set. After you know this information, set the default card with the following command as the `root` user:

```
cat > /etc/asound.conf << "EOF"
# Begin /etc/asound.conf

defaults.pcm.card 1
defaults.ctl.card 1

# End /etc/asound.conf
EOF
```

Contents

Installed Programs: aconnect, alsactl, alsaloop, alsamixer, alsatplg, alsaucm, alsa-info.sh, amidi, amixer, aplay, aplaymidi, arecord (symlink), arecordmidi, aseqdump, aseqnet, axfer, iecset, and speaker-test

Installed Libraries: None

Installed Directories: /usr/share/sounds/alsa and /var/lib/alsa

Short Descriptions

<code>aconnect</code>	is a utility for connecting and disconnecting two existing ports in the ALSA sequencer system
<code>alsactl</code>	is used to control advanced settings for ALSA sound card drivers. Use this if alsamixer cannot utilize all of your sound card's features
<code>alsaloop</code>	allows creation of a PCM loopback between a PCM capture device and a PCM playback device
<code>alsamixer</code>	is an Ncurses based mixer program for use with the ALSA sound card drivers
<code>alsatplg</code>	is a utility used to compile topology configuration files into binary files for kernel drivers
<code>alsaucm</code>	allows applications to access the hardware in an abstracted manner
<code>amidi</code>	is used to read from and write to ALSA RawMIDI ports
<code>amixer</code>	allows command-line control of the mixers for the ALSA sound card drivers
<code>aplay</code>	is a command-line soundfile player for the ALSA sound card drivers
<code>aplaymidi</code>	is a command-line utility that plays the specified MIDI file(s) to one or more ALSA sequencer ports
<code>arecord</code>	is a command-line soundfile recorder for the ALSA sound card drivers
<code>arecordmidi</code>	is a command-line utility that records a standard MIDI file from one or more ALSA sequencer ports
<code>aseqdump</code>	is a command-line utility that prints the sequencer events it receives as text
<code>aseqnet</code>	is an ALSA sequencer client which sends and receives event packets over a network
<code>axfer</code>	is a command-line recorder and player used to transfer audio frames between sound devices and files
<code>iecset</code>	is a small utility to set or dump the IEC958 (or so-called "S/PDIF") status bits of the specified sound card via the ALSA control API
<code>speaker-test</code>	is a command-line speaker test tone generator for ALSA

alsa-tools-1.2.11

Introduction to ALSA Tools

The ALSA Tools package contains advanced tools for certain sound cards.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.alsa-project.org/files/pub/tools/alsa-tools-1.2.11.tar.bz2>
- Download MD5 sum: bc5f5e5689f46a9d4a0b85dc6661732c
- Download size: 1.8 MB
- Estimated disk space required: 18 MB
- Estimated build time: 0.3 SBU

ALSA Tools Dependencies

Required

[alsa-lib-1.2.12](#)

Optional

[GTK+-3.24.43](#) (to build `hdajackretask`), [FLTK-1.3.9](#) (to build `hdspconf` and `hdspmixer`), and [GTK+-2](#) (to build `echomixer`, `envy24control`, and `rmedigicontrol`)

Installation of ALSA Tools

Note

When installing multiple packages in a script, the installation needs to be done as the root user. There are three general options that can be used to do this:

1. Run the entire script as the root user (not recommended).
2. Use the `sudo` command from the [Sudo-1.9.15p5](#) package.
3. Use `su -c "command arguments"` (quotes required) which will ask for the root password for every iteration of the loop.

One way to handle this situation is to create a short `bash` function that automatically selects the appropriate method. Once the command is set in the environment, it does not need to be set again.

```
as_root()
{
    if [ $EUID = 0 ];      then $*
    elif [ -x /usr/bin/sudo ]; then sudo $*
    else
        su -c \\\"$*\\\"
    fi
}

export -f as_root
```

First, start a subshell that will exit on error:

```
bash -e
```

Now, remove some tools that need Qt2 or 3 or [GTK+-2](#), and also two unneeded files (for the BLFS instructions below):

```
rm -rf qlo10k1 echomixer envy24control rmedigicontrol Makefile gitcompile
```

The ALSA Tools package is only needed by those with advanced requirements for their sound card. The tools can be built all together at once, but if only a subset is needed, you need to `cd` into the directory of each tool you wish to compile and run the commands. Here, we present instructions to build all tools.

Install all ALSA Tools by running the following commands:

```
for tool in *
do
  case $tool in
    seq )
      tool_dir=seq/sbiload
    ;;
    * )
      tool_dir=$tool
    ;;
  esac

  pushd $tool_dir
  ./configure --prefix=/usr
  make
  as_root make install
  as_root /sbin/ldconfig
  popd

done
unset tool tool_dir
```

Finally, exit the shell that was started earlier:

```
exit
```

Contents

Installed Programs: as10k1, cspctl, dl10k1, hda-verb, hdajackretask, hdajacksensetest, hdspconf, hdsloader, hdspmixer, hwmixvolume, init_audigy, init_audigy_eq10, init_live, lo10k1, ld10k1, ld10k1d, mixartloader, pcxhrloader, sbiload, sscape_ctl, us428control, usx2yloader, and vxloader

Installed Library: liblo10k1.so

Installed Directories: /etc/hotplug, /usr/include/lo10k1, /usr/share/ld10k1, and /usr/share/sounds

Short Descriptions

as10k1	is an assembler for the emu10k1 DSP chip present in the Creative SB Live, PCI 512, and emu APS sound cards. It is used to make audio effects such as a flanger, chorus or reverb
cspctl	is an SB16/AWE32 Creative Signal Processor (ASP/CSP) control program
hdajackretask	is a GUI to make it easy to retask your jacks - e.g., turn your Mic jack into an extra Headphone, or make them both line outs and connect them to your surround receiver
hda-verb	is a small program to send HD-Audio commands to the given ALSA hwdep device on the hd-audio interface
hdspconf	is a GUI to control the Hammerfall HDSP Alsa Settings. Up to four hdsp cards are supported
hdsloader	is used to load the firmware required by the Hammerfall HDSP sound cards
hdspmixer	is the Linux equivalent of the Totalmix application from RME. It is a tool to control the advanced routing features of the RME Hammerfall DSP soundcard series
hwmixvolume	allows you to control the volume of individual streams on sound cards that use hardware mixing
init_audigy*	are tools used to initialize Creative Sound Blaster Audigy-series cards
init_live	is a tool used to initialize Creative Sound Blaster Live cards
ld10k1	is the server of a EMU10K{1,2} patch loader for ALSA
lo10k1	is the client of a EMU10K{1,2} patch loader for ALSA
dl10k1	loads config dumps generated by <code>ld10k1</code> and <code>1d10k1</code>
1d10k1d	is an init script for the <code>ld10k1</code> patch loader
mixartloader	is a helper program to load the firmware binaries onto the Digigram's miXart board sound drivers. The snd-mixart module requires this program. These drivers don't work properly until the required firmware files are loaded, i.e. no PCM nor mixer devices will appear
pcxhrloader	is a helper program to load the firmware binaries onto Digigram's pcxhr compatible board sound drivers. The snd-pcxhr module requires this program. These drivers don't work properly until certain firmware files are loaded, i.e. no PCM nor mixer devices will appear
sbiload	is an OPL2/3 FM instrument loader for the ALSA sequencer
sscape_ctl	is an ALSA SoundScape control utility
us428control	is a Tascam US-428 control program

usx2yloader	is a helper program to load the 2nd Phase firmware binaries onto the Tascam USX2Y USB sound cards. It has proven to work so far for the US122, US224 and US428. The snd-usb-usx2y module requires this program
vxloader	is a helper program to load the firmware binaries onto the Digigram's VX-board sound drivers. The snd-vx222, snd-vxpocket, and snd-vxp440 modules require this program. These drivers don't work properly until certain firmware files are loaded, i.e. no PCM nor mixer devices will appear

alsa-firmware-1.2.4

Introduction to ALSA Firmware

The ALSA Firmware package contains firmware for certain sound cards.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.alsa-project.org/files/pub/firmware/alsa-firmware-1.2.4.tar.bz2>
- Download MD5 sum: ee6c1d24a1a4ac1d86992b408ed710a2
- Download size: 4.9 MB
- Estimated disk space required: 39 MB
- Estimated build time: less than 0.1 SBU

ALSA Firmware Dependencies

Required

[alsa-tools-1.2.11](#)

Optional

[AS31](#) (for rebuilding the firmware from source)

Installation of ALSA Firmware

The ALSA Firmware package is only needed by those with advanced requirements for their sound card. See the README for configure options.

Install ALSA Firmware by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: Several directories in /lib/firmware and /usr/share/alsa/firmware

AudioFile-0.3.6

Introduction to AudioFile

The AudioFile package contains the audio file libraries and two sound file support programs useful to support basic sound file formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/audiofile/0.3/audiofile-0.3.6.tar.xz>
- Download MD5 sum: 235dde14742317328f0109e9866a8008
- Download size: 520 KB
- Estimated disk space required: 18 MB
- Estimated build time: 0.6 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/audiofile-0.3.6-consolidated_patches-1.patch

AudioFile Dependencies

Required

[alsa-lib-1.2.12](#)

Recommended

[FLAC-1.4.3](#)

Optional

[asciidoc-10.2.1](#) and [Valgrind-3.23.0](#)

Installation of AudioFile

Install AudioFile by running the following commands:

```
patch -Np1 -i ../../audiofile-0.3.6-consolidated_patches-1.patch &&
autoreconf -fiv &&

./configure --prefix=/usr --disable-static &&
make
```

Now, as the `root` user:

```
make install
```

To test the results, issue: `make check`. Note that the tests will fail if the `--disable-static` option is used and tests are executed before `make install`. You have three options:

- configure without `--disable-static`, run the tests, but do not install, then start a fresh build using `--disable-static` just for installing the package.
- configure with `--disable-static`, but only run the tests after the package is installed.
- configure with `--disable-static`, but only run the tests after a `DESTDIR` install.

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: sfconvert and sfinfo

Installed Libraries: libaudiofile.so

Installed Directories: None

Short Descriptions

<code>sfinfo</code>	displays the sound file format, audio encoding, sampling rate and duration for audio formats supported by this library
---------------------	--

<code>sfconvert</code>	converts sound file formats where the original format and destination format are supported by this library
<code>libaudiofile.so</code>	contains functions used by programs to support AIFF, AIFF-compressed, Sun/NeXT, WAV and BIC audio formats

FAAC-1_30

Introduction to FAAC

FAAC is an encoder for a lossy sound compression scheme specified in MPEG-2 Part 7 and MPEG-4 Part 3 standards and known as Advanced Audio Coding (AAC). This encoder is useful for producing files that can be played back on an iPod. Moreover, the iPod does not understand other sound compression schemes used in video files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://github.com/knik0/faac/archive/1_30/faac-1_30.tar.gz
- Download MD5 sum: 8d61e6d55088e599aa91532d5e6995b0
- Download size: 240 KB
- Estimated disk space required: 4.7 MB
- Estimated build time: less than 0.1 SBU

Installation of FAAC

Install FAAC by running the following commands:

```
./bootstrap
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite. However, basic functionality can be tested by encoding a sample WAV file (the sample file is installed by the [alsa-utils-1.2.12](#) package):

```
./frontend/faac -o Front_Left.mp4 /usr/share/sounds/alsa/Front_Left.wav
```

Then, decode the result using the `faad` program from the [FAAD2-2.11.1](#) package and play back the decoded file (requires the `aplay` program from the [alsa-utils-1.2.12](#) package):

```
faad Front_Left.mp4
aplay Front_Left.wav
```

`aplay` should identify the file as Signed 16 bit Little Endian, Rate 48000 Hz, Stereo, and you should hear the words "front left."

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Other AAC encoders

The quality of FAAC is not up to par with the best AAC encoders currently available. Also, it only supports AAC and not High Efficiency AAC (also known as aacPlus), which provides better quality at low bitrates by means of using the "spectral band replication" technology. One example of an alternative program for producing AAC and HE-AAC streams is:

- [3GPP Enhanced aacPlus general audio codec](#): available in the source form, can encode only HE-AAC up to 48 kbps out of the box, but the maximum bitrate can be changed by editing the tuning table in the `FloatFR_sbrenclib/src/sbr_main.c` file.

Note, however, that the iPod supports only the Low Complexity AAC profile, which is the default in FAAC, but is completely unavailable in the 3GPP encoder.

Contents

Installed Program: faac

Installed Libraries: libfaac.so and libmp4v2.so

Installed Directories: None

Short Descriptions

faac	is a command-line AAC encoder
libfaac.so	contains functions for encoding AAC streams
libmp4v2.so	contains functions for creating and manipulating MP4 files

FAAD2-2.11.1

Introduction to FAAD2

FAAD2 is a decoder for a lossy sound compression scheme specified in MPEG-2 Part 7 and MPEG-4 Part 3 standards and known as Advanced Audio Coding (AAC).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/knik0/faad2/archive/2.11.1/faad2-2.11.1.tar.gz>
- Download MD5 sum: f85b2514c4fb2f87d22a3bc879d83277
- Download size: 642 KB
- Estimated disk space required: 9.0 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Sample AAC file: <https://www.nch.com.au/acm/sample.aac> (7 KB)

FAAD2 Dependencies

Required

[CMake-3.30.2](#)

Installation of FAAD2

Install FAAD2 by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_SHARED_LIBS=ON \
      ...
make
```

This package does not come with a test suite. However, basic functionality can be tested by decoding the sample AAC file:

```
./faad -o sample.wav ../../sample.aac
```

This should display a copyright message and the following information about the sample file:

```
sample.aac file info:
ADTS, 4.608 sec, 13 kbps, 16000 Hz
```

```
-----
| Config: 2 Ch      |
-----
| Ch | Position    |
-----
| 00 | Left front  |
```

```
| 01 | Right front |
```

Now play the result (requires the `aplay` program from the [alsa-utils-1.2.12](#) package):

```
aplay sample.wav
```

`aplay` should identify the file as Signed 16 bit Little Endian, Rate 16000 Hz, Stereo, and you should hear some piano notes.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: faad

Installed Library: libfaad.so and libfaad_drm.so

Installed Directories: None

Short Descriptions

`faad` is a command-line utility for decoding AAC and MP4 files
`libfaad.so` contains functions for decoding AAC streams

fdk-aac-2.0.3

Introduction to fdk-aac

The fdk-aac package provides the Fraunhofer FDK AAC library, which is purported to be a high quality Advanced Audio Coding implementation.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/opencore-amr/fdk-aac-2.0.3.tar.gz>
- Download MD5 sum: f43e593991caefcde509ad837d3301bd
- Download size: 2.8 MB
- Estimated disk space required: 39 MB
- Estimated build time: 0.6 SBU (Using parallelism=4)

Installation of fdk-aac

Install fdk-aac by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libfdk-aac.so

Installed Directory: /usr/include/fdk-aac

Short Descriptions

libfdk-aac.so provides the functions used to encode audio in AAC format

FLAC-1.4.3

Introduction to FLAC

FLAC is an audio CODEC similar to MP3, but lossless, meaning that audio is compressed without losing any information.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/flac/flac-1.4.3.tar.xz>
- Download MD5 sum: 7cab20b46e0bc859234bf5ba7da81625
- Download size: 1.0 MB
- Estimated disk space required: 22 MB (additional 163 MB to run the test suite)
- Estimated build time: 0.2 SBU (additional 0.6 SBU to run the test suite)

FLAC Dependencies

Optional

[libogg-1.3.5](#), [DocBook-utils-0.6.14](#), [Doxygen-1.12.0](#), and [Valgrind-3.23.0](#)

Installation of FLAC

Install FLAC by running the following commands:

```
./configure --prefix=/usr \
            --disable-thorough-tests \
            --docdir=/usr/share/doc/flac-1.4.3 \
            &&
make
```

To test the results, issue: `make check`. Note that if you passed the `--enable-exhaustive-tests` and `--enable-valgrind-testing` parameters to `configure` and then run the test suite, it will take a very long time (up to 300 SBUs) and use about 375 MB of disk space.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-thorough-tests`: This parameter is used so that the test suite will complete in a reasonable amount of time. Remove it if you desire more extensive tests.

Contents

Installed Programs: flac and metaflac

Installed Libraries: libFLAC.so and libFLAC++.so

Installed Directories: /usr/include/FLAC, /usr/include/FLAC++ and /usr/share/doc/flac-1.4.3

Short Descriptions

`flac` is a command-line utility for encoding, decoding and converting FLAC files

`metaflac` is a program for listing, adding, removing, or editing metadata in one or more FLAC files

`libFLAC{,++}.so` these libraries provide native FLAC and Ogg FLAC C/C++ APIs for programs utilizing FLAC

frei0r-plugins-1.8.0

Introduction to Frei0r-plugins

Frei0r is a minimalistic plugin API for video effects. Note that the 0 in the name is a zero, not a capital letter o.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.dyne.org/frei0r/releases/frei0r-plugins-1.8.0.tar.gz>
- Download MD5 sum: 45ffe53925ce0a90ce1d838c05e0a3c0
- Download size: 804 KB
- Estimated disk space required: 40 MB
- Estimated build time: 0.5 SBU

Frei0r-plugins Dependencies

Recommended

[gavl-1.4.0](#)

Optional

[Doxygen-1.12.0](#) and [opencv-4.10.0](#)

Installation of Frei0r-plugins

Install Frei0r by running the following commands:

```
mkdir -vp build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -W no-dev ..          &&

make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`-D WITHOUT_OPENCV=TRUE`: This option instructs the make procedure to disable building plugins dependent upon opencv.

`-D WITHOUT_GAVL=TRUE`: This option instructs the make procedure to disable building plugins dependent upon gavl.

Contents

Installed Program: None

Installed Libraries: Over 130 video effects plugins

Installed Directories: /usr/lib/frei0r-1

gavl-1.4.0

Introduction to Gavl

Gavl is short for Gmerlin Audio Video Library. It is a low level library that handles the details of audio and video formats like colorspace, samplers, multichannel configurations etc. It provides standardized definitions for those formats as well as container structures for carrying audio samples or video images inside an application.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/gmerlin/gavl-1.4.0.tar.gz>
- Download MD5 sum: 2752013a817fbc43ddf13552215ec2c0
- Download size: 4.4 MB
- Estimated disk space required: 50 MB
- Estimated build time: 0.8 SBU

Gavl Dependencies

Required

[libpng-1.6.43](#)

Optional

[Doxygen-1.12.0](#)

Installation of Gavl

First, fix an issue with gcc-14 and later:

```
sed -i "/stdio/a #include <string.h>" src/fill_test.c
```

Now, install Gavl by running the following commands:

```
LIBS=-lm \
./configure --prefix=/usr \
--without-doxygen \
--with-cpuflags=none \
--docdir=/usr/share/doc/gavl-1.4.0 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`LIBS=-lm`: This variable ensures the math library is searched when linking.

`--without-doxygen`: This switch disables use of Doxygen. Omit if Doxygen is installed and you wish to build the API documentation.

`--with-cpuflags=none`: This switch fixes a problem identifying capabilities of the system architecture.

Contents

Installed Program: None

Installed Libraries: libgavl.so

Installed Directories: /usr/include/gavl and /usr/share/doc/gavl-1.4.0

Short Descriptions

`libgavl.so` is the Gmerlin Audio Video Library

gstreamer-1.24.7

Introduction to gstreamer

gstreamer is a streaming media framework that enables applications to share a common set of plugins for tasks such as video encoding and decoding, audio encoding and decoding, audio and video filters, audio visualisation, web streaming and anything else that streams in real-time or otherwise. This package only provides base functionality and libraries. You may need at least [gst-plugins-base-1.24.7](#) and one of Good, Bad, Ugly or Libav plugins.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gstreamer.freedesktop.org/src/gstreamer/gstreamer-1.24.7.tar.xz>
- Download MD5 sum: 5abf79978005ae14a8bc7db1ac3ce4ba
- Download size: 1.8 MB
- Estimated disk space required: 56 MB (with tests)
- Estimated build time: 0.3 SBU (Using parallelism=4; with tests)

gststreamer Dependencies

Required

[GLib-2.80.4](#) (GObject Introspection required for GNOME)

Optional

[GTK+-3.24.43](#) (for examples), [gsl-2.8](#) (used by one test if installed), [libunwind-1.6.2](#), [Valgrind-3.23.0](#), [bash-completion](#), [hotdoc](#), and [libdw](#)

Installation of gstreamer

Install gstreamer by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gst_debug=false &&
ninja
```

To test the results, issue: `ninja test`.

Caution

If you are reinstalling gstreamer from a previous version, it is best if you remove the prior version, including plugins, before installing the new version. If there is a mixture of versions installed, using processes may hang or not work properly. As the `root` user:

```
rm -rf /usr/bin/gst-* /usr/{lib,libexec}/gstreamer-1.0
```

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: `gst-inspect-1.0`, `gst-launch-1.0`, `gst-stats-1.0`, `gst-tester-1.0`, and `gst-typefind-1.0`

Installed Libraries: `libgstbase-1.0.so`, `libgstcheck-1.0.so`, `libgstcontroller-1.0.so`, `libgstnet-1.0.so`, and `libgststreamer-1.0.so`

Installed Directories: `/usr/{include,lib,libexec,share}/gstreamer-1.0`

Short Descriptions

<code>gst-inspect-1.0</code>	is a tool that prints out information on available gstreamer plugins, information about a particular plugin, or information about a particular element
------------------------------	--

<code>gst-launch-1.0</code>	is a tool that builds and runs basic gstreamer pipelines
<code>gst-stats-1.0</code>	is a tool used to gather statistics about gstreamer operations
<code>gst-tester-1.0</code>	runs a test plan in TAP compatible format while integrating with the meson test harness
<code>gst-typefind-1.0</code>	uses the gstreamer type finding system to determine the relevant gstreamer plugin to parse or decode files, and the corresponding MIME type
<code>libgstbase-1.0.so</code>	provides some base classes to be extended by elements and utility classes that are most useful for plugin developers
<code>libgstcheck-1.0.so</code>	provides functionality for writing unit tests that use the check framework
<code>libgstcontroller-1.0.so</code>	provides functionality to animate element properties over time
<code>libgstnet-1.0.so</code>	provides network elements and objects
<code>libgststreamer-1.0.so</code>	provides all of the core gstreamer services, including initialization, plugin management and types, as well as the object hierarchy that defines elements and bins, along with some more specialized elements

gst-plugins-base-1.24.7

Introduction to GStreamer Base Plug-ins

The GStreamer Base Plug-ins is a well-groomed and well-maintained collection of GStreamer plug-ins and elements, spanning the range of possible types of elements one would want to write for GStreamer. You will need at least one of Good, Bad, Ugly or Libav plugins for GStreamer applications to function properly.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gstreamer.freedesktop.org/src/gst-plugins-base/gst-plugins-base-1.24.7.tar.xz>
- Download MD5 sum: 8a7f8668418d0c8c917bb92218047f9f
- Download size: 2.3 MB
- Estimated disk space required: 88 MB (with tests)
- Estimated build time: 1.0 SBU (Using parallelism=4; with tests)

GStreamer Base Plug-ins Dependencies

Required

[gstreamer-1.24.7](#)

Recommended

[alsa-lib-1.2.12](#), [CDParanoia-III-10.2](#) (for building the CDDA plugin), [GLib-2.80.4](#) (with GObject Introspection), [ISO Codes-4.16.0](#), [libgudev-238](#), [libjpeg-turbo-3.0.1](#), [libogg-1.3.5](#), [libpng-1.6.43](#), [libvorbis-1.3.7](#), [Mesa-24.1.5](#), [Pango-1.54.0](#), [wayland-protocols-1.36](#), and [Xorg Libraries](#)

Optional

[graphene-1.10.8](#), [GTK+-3.24.43](#) (for examples), [Opus-1.5.2](#), [gt5-components-5.15.14](#) (for examples), [SDL2-2.30.6](#), [Valgrind-3.23.0](#), [hotdoc](#), [libtheora](#), [libvisual](#), [Orc](#), and [Tremor](#)

Installation of GStreamer Base Plug-ins

Note

If you need a plugin for a given dependency, that dependency needs to be installed before this package.

Install GStreamer Base Plug-ins by running the following commands:

```
mkdir build &&
cd      build &&

meson setup ..          \
```

```
--prefix=/usr      \
--buildtype=release \
--wrap-mode=nodownload &&
ninja
```

To test the results, issue: `ninja test`. The tests require an X terminal running, or all of the GL tests will fail. Five tests may produce timeouts on some systems depending on their graphics hardware and speed. One test, `elements_apps`, is known to fail on some systems.

Note

When installing, the build process does some additional linking. If you do not have Xorg in /usr, the LIBRARY_PATH variable needs to be defined for the root user. If using sudo to assume root, use the -E option to pass your current environment variables for the install process.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`--wrap-mode=nodownload`: This switch prevents `meson` from downloading any optional dependency which is not installed on the system.

Contents

Installed Programs: `gst-device-monitor-1.0`, `gst-discoverer-1.0`, and `gst-play-1.0`

Installed Libraries: `libgstallocators-1.0.so`, `libgstapp-1.0.so`, `libgstaudio-1.0.so`, `libgstfft-1.0.so`, `libgstgl-1.0.so`, `libgstrtbutils-1.0.so`, `libgsttriff-1.0.so`, `libgstrtp-1.0.so`, `libgstrtsp-1.0.so`, `libgstsdp-1.0.so`, `libgsttag-1.0.so`, `libgstvideo-1.0.so`, and several plugins under `/usr/lib/gstreamer-1.0`

Installed Directories: `/usr/include/gstreamer-1.0/gst/{allocators,app,audio,fft,gl,pbutils}`, `/usr/include/gstreamer-1.0/gst/{riff,rtp,rtsp,sdp,tag,video}`, and `/usr/share/gst-plugins-base`

Short Descriptions

<code>gst-device-monitor-1.0</code>	is a command line tool that can be used to test GStreamer's device monitoring functionality
<code>gst-discoverer-1.0</code>	is a tool that can be used to print basic metadata and stream information about a media file
<code>gst-play-1.0</code>	is a command line tool that can be used to test basic playback using the playbin element

gst-plugins-good-1.24.7

Introduction to GStreamer Good Plug-ins

The GStreamer Good Plug-ins is a set of plug-ins considered by the GStreamer developers to have good quality code, correct functionality, and the preferred license (LGPL for the plug-in code, LGPL or LGPL-compatible for the supporting library). A wide range of video and audio decoders, encoders, and filters are included.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gstreamer.freedesktop.org/src/gst-plugins-good/gst-plugins-good-1.24.7.tar.xz>
- Download MD5 sum: 48477b1fe91522b6a0b733c22a780d78
- Download size: 2.8 MB
- Estimated disk space required: 105 MB (with tests)
- Estimated build time: 0.6 SBU (Using parallelism=4; with tests)

GStreamer Good Plug-ins Dependencies

Required

[gst-plugins-base-1.24.7](#)

Recommended

[Cairo-1.18.0](#), [FLAC-1.4.3](#), [gdk-pixbuf-2.42.12](#), [LAME-3.100](#), [libsoup-2.74.3](#), [libsoup-3.4.4](#), [libvpx-1.14.1](#), [mpg123-1.32.7](#), [NASM-2.16.03](#), and [PulseAudio-17.0](#)

Optional

[AAlib-1.4rc5](#), [GTK+-3.24.43](#) (for examples), [libdv-1.0.0](#), [qt5-components-5.15.14](#), [Qt-6.7.2](#), [Speex-1.2.1](#), [taglib-2.0.1](#), [Valgrind-3.23.0](#), [v4l-utils-1.28.1](#), [Wayland-1.23.0](#), [alsa-oss](#), [hotdoc](#), [JACK](#), [libcaca](#), [libavc1394](#), [libiec61883](#), [libraw1394](#), [libshout](#), [Orc](#), [TwoLame](#), and [WavPack](#)

Installation of GStreamer Good Plug-ins

Note

If you need a plugin for a given dependency, that dependency needs to be installed before this package.

Install GStreamer Good Plug-ins by running the following commands:

```
mkdir build &&
cd      build &&

meson setup ..          \
    --prefix=/usr        \
    --buildtype=release &&
ninja
```

To test the results, issue: `ninja test`. One test, `elements_flvmux`, is known to fail on some systems.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Libraries: Several plugins under /usr/lib/gstreamer-1.0

Installed Directories: /usr/share/gstreamer-1.0/presets

gst-plugins-bad-1.24.7

Introduction to GStreamer Bad Plug-ins

The GStreamer Bad Plug-ins package contains a set of plug-ins that aren't up to par compared to the rest. They might be close to being good quality, but they're missing something - be it a good code review, some documentation, a set of tests, a real live maintainer, or some actual wide use.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gstreamer.freedesktop.org/src/gst-plugins-bad/gst-plugins-bad-1.24.7.tar.xz>
- Download MD5 sum: a0a2771d1ae1203004d58d80bc6fcfd16
- Download size: 6.7 MB

- Estimated disk space required: 171 MB (with tests)
- Estimated build time: 0.7 SBU (Using parallelism=4; with tests)

GStreamer Bad Plug-ins Dependencies

Required

[gst-plugins-base-1.24.7](#)

Recommended

[libdvdread-6.1.3](#), [libdvdnav-6.1.1](#), [libva-2.22.0](#), and [SoundTouch-2.3.3](#)

Optional

[BlueZ-5.77](#), [cURL-8.9.1](#), [FAAC-1_30](#), [FAAD2-2.11.1](#), [fdk-aac-2.0.3](#), [GTK+-3.24.43](#) (for examples), [gst-plugins-good-1.24.7](#) (for one test), [JSON-GLib-1.8.0](#), [Little CMS-2.16](#), [libaom-3.9.1](#), [libass-0.17.3](#), [libexif-0.6.24](#) (for one test), [librsvg-2.58.3](#), [libsoup-2.74.3](#) (for one test), [libsndfile-1.2.2](#), [libssh2-1.11.0](#), [libusb-1.0.27](#), [libwebp-1.4.0](#), [libxkbcommon-1.7.0](#), [neon-0.33.0](#), [Nettle-3.10](#) or [libgcrypt-1.11.0](#) (for SSL support in the hls plugin, if both are not installed OpenSSL will be used instead), [opencv-4.10.0](#) (with additional modules), [OpenJPEG-2.5.2](#), [Opus-1.5.2](#), [gencode-4.1.1](#), [SBC-2.0](#), [sdl12-compat-1.2.68](#), [Valgrind-3.23.0](#), both [Vulkan-Loader-1.3.294](#) and [glslc](#) (for Vulkan plugin), [Wayland-1.23.0](#) ([GTK+-3.24.43](#) must have been compiled with wayland support), [x265-3.6](#), [bs2b](#), [Chromaprint](#), [dssim](#), [Flite](#), [FluidSynth](#), [Game Music Emu](#), [GSM](#), [hotdoc](#), [LADSPA](#), [IdacBT](#), [libajantv2](#), [libavtp](#), [libdc1394-2](#), [libdca](#), [libde265](#), [libkate](#), [libmfx](#), [libmms](#), [libmodplug](#), [libnice](#), [libofa](#), [libopenempt](#), [libopenni](#), [libsrt](#), [lilv](#), [LRDF](#), [ltc-tools](#), [microdns](#), [MJPEG Tools](#), [mplex2](#), [musepack](#), [onnxruntime](#), [OpenAL](#), [OpenEXR](#), [OpenH264](#), [Orc](#), [rtmpdump](#), [spandsp](#), [Srt](#), [svthevcenc](#), [VO AAC](#), [VO AMRWB](#), [WildMidi](#), [WPE-WebKit](#), [WPEBackend-fdo](#), [ZBAR](#), [ZVBI](#), and [xzing](#)

Installation of GStreamer Bad Plug-ins

Note

If you need a plugin for a given dependency, that dependency needs to be installed before this package.

Install GStreamer Bad Plug-ins by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gpl=enabled &&
ninja
```

To test the results, issue: `ninja test`. Several tests need a terminal emulator in a graphical session. Two tests, `elements_vapostproc` and `elements_dash_mpd`, are known to fail. If [gst-plugins-good-1.24.7](#) is not installed, two tests, `elements_rtpsrc` and `elements_rtpsink`, will fail.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gpl=enabled`: Without this switch, plugins with dependencies on (A)GPL licensed libraries are not built.

Contents

Installed Programs: `gst-transcoder-1.0` and `playout`

Installed Libraries: `libgstadaptivedemux-1.0.so`, `libgstanalytics-1.0.so`, `libgstbadaudio-1.0.so`, `libgstbasecamerabinsrc-1.0.so`, `libgstcuda-1.0.so`, `libgstcodecparsers-1.0.so`, `libgstcodecs-1.0.so`, `libgstdxva-1.0.so`, `libgstinsertbin-1.0.so`, `libgstisoff-1.0.so`, `libgstmpegts-1.0.so`, `libgstmse-1.0.so`, `libgstphotography-1.0.so`, `libgstplay-1.0.so`, `libgstplayer-1.0.so`, `libgstscpp-1.0.so`, `libgsttranscoder-1.0.so`,

libgsturidownloader-1.0.so, libgstvfa-1.0.so, libgstwayland-1.0.so, libgstwebrtc-1.0.so, and several plugins under /usr/lib/gstreamer-1.0

Installed Directories: /usr/include/gststreamer-1.0/gst/{audio,basecamerabinsrc,cuda}, /usr/include/gststreamer-1.0/gst/{codecparsers,insertbin,interfaces}, /usr/include/gststreamer-1.0/gst/{isoff,mpegs,play,player,sctp} /usr/include/gststreamer-1.0/gst/{transcoder,uridownloader,va}, and /usr/include/gststreamer-1.0/gst/{wayland,webrtc}

Short Descriptions

<code>gst-transcoder-1.0</code>	is used to transcode a stream into a different format
<code>playout</code>	is an example application used to sequentially play a list of audio-video files

gst-plugins-ugly-1.24.7

Introduction to GStreamer Ugly Plug-ins

The GStreamer Ugly Plug-ins is a set of plug-ins considered by the GStreamer developers to have good quality and correct functionality, but distributing them might pose problems. The license on either the plug-ins or the supporting libraries might not be how the GStreamer developers would like. The code might be widely known to present patent problems.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gstreamer.freedesktop.org/src/gst-plugins-ugly/gst-plugins-ugly-1.24.7.tar.xz>
- Download MD5 sum: 880ffe8dfee6f1dce2127190dbc3841d
- Download size: 224 KB
- Estimated disk space required: 8.4 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

GStreamer Ugly Plug-ins Dependencies

Required

[gst-plugins-base-1.24.7](#)

Recommended

[liba52-0.8.0](#) (needed to play DVD's), [libdvread-6.1.3](#), and [x264-20240812](#)

Optional

[libmpeg2-0.5.1](#), [libcdio-2.1.0](#) (for CD-ROM drive access), [Valgrind-3.23.0](#), [hotdoc](#), [libsidplay](#), and [Orc](#)

Installation of GStreamer Ugly Plug-ins

Note

If you need a plugin for a given dependency, that dependency needs to be installed before this package.

Install GStreamer Ugly Plug-ins by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gpl=enabled &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D gpl=enabled`: Without this switch, plugins with dependencies on (A)GPL licensed libraries are not built.

Contents

Installed Programs: None

Installed Libraries: Several plugins under /usr/lib/gstreamer-1.0

Installed Directories: None

gst-libav-1.24.7

Introduction to GStreamer Libav

The GStreamer Libav package contains GStreamer plugins for Libav (a fork of FFmpeg).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gstreamer.freedesktop.org/src/gst-libav/gst-libav-1.24.7.tar.xz>
- Download MD5 sum: be00cf9f9c8ca13791e6193f584213b6
- Download size: 208 KB
- Estimated disk space required: 14 MB
- Estimated build time: less than 0.1 SBU

GStreamer Libav Dependencies

Required

[FFmpeg-7.0.2](#) and [gst-plugins-base-1.24.7](#)

Recommended

[yasm-1.3.0](#)

Optional

[hotdoc](#)

Installation of GStreamer Libav

Install GStreamer Libav by running the following commands:

```
mkdir build &&
cd build &&

meson setup ..           \
    --prefix=/usr          \
    --buildtype=release &&
ninja
```

To run the tests, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

--buildtype=release: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Library: libgstlibav.so in /usr/lib/gstreamer-1.0

Installed Directory: None

id3lib-3.8.3

Introduction to id3lib

id3lib is a library for reading, writing and manipulating id3v1 and id3v2 multimedia data containers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/id3lib/id3lib-3.8.3.tar.gz>
- Download MD5 sum: 19f27ddd2dda4b2d26a559a4f0f402a7
- Download size: 932 KB
- Estimated disk space required: 16 MB
- Estimated build time: 0.2 SBU

Additional Downloads

- Required patches: https://www.linuxfromscratch.org/patches/blfs/12.2/id3lib-3.8.3-consolidated_patches-1.patch

Installation of Id3lib

Install id3lib by running the following commands:

```
patch -Npl1 -i ../id3lib-3.8.3-consolidated_patches-1.patch &&
libtoolize --force &&
aclocal &&
autoconf &&
automake --add-missing --copy &&
./configure --prefix=/usr --disable-static &&
make
```

This packages does not come with a working test suite.

Now, as the `root` user:

```
make install &&
cp doc/man/* /usr/share/man/man1 &&
install -v -m755 -d /usr/share/doc/id3lib-3.8.3 &&
install -v -m644 doc/*.{gif,jpg,png,ico,css,txt,php,html} \
/usr/share/doc/id3lib-3.8.3
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: id3convert, id3cp, id3info, and id3tag

Installed Library: libid3.so

Installed Directories: /usr/include/id3 and /usr/share/doc/id3lib-3.8.3

Short Descriptions

<code>id3convert</code>	converts between id3v1/v2 tagging formats
<code>id3cp</code>	extracts id3v1/v2 tags from digital audio files
<code>id3info</code>	prints id3v1/v2 tag contents
<code>id3tag</code>	is a utility for editing id3v1/v2 tags
<code>libid3.so</code>	provides functions for the id3v1/v2 tag editing programs as well as other external programs and libraries

intel-media-24.2.5

Introduction to intel-media

The intel-media package provides a VA API driver for Intel GPUs that are provided with Broadwell CPUs and higher. This includes support for a variety of codecs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/intel/media-driver/archive/intel-media-24.2.5.tar.gz>
- Download MD5 sum: c602d9a3ebbeb2ad8a5239dc5951c65b
- Download size: 25 MB
- Estimated disk space required: 2.1 GB (360 MB installed with a single GPU model)
- Estimated build time: 3.8 SBU (with parallelism=4 and a single GPU model)

Note

The tarball `intel-media-24.2.5.tar.gz` will extract to the directory `media-driver-intel-media-24.2.5`.

intel-media Dependencies

Required

[CMake-3.30.2](#), [Intel-gmmlib-22.4.1](#), [libva-2.22.0](#), and [Xorg build environment](#)

Kernel Configuration

Enable the following options in the kernel configuration. Recompile the kernel if necessary:

```
Device Drivers --->
  Graphics support --->
    <*/M> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
      ...
      [DRM]
    <*/M> Intel 8xx/9xx/G3x/G4x/HD Graphics
          [DRM_I915]
```

Installation of intel-media

Note

This package takes a long time to build because it compiles code specific to each individual generation of Intel GPUs and for a variety of media codecs.

If you know the model of your Intel GPU, you can pass the `-D{GEN{8,9,11,12},MTL,ARL}=OFF` option to the `cmake` command but leaving the option for your GPU out. Note that the "GEN" number here is the generation of the GPU, not the CPU. For example, with an Intel Core i7-1065G7 CPU shipping a 11th-generation Intel GPU, the `-D{GEN{8,9,12},MTL,ARL}=OFF` option can be used so the code specific to the other generations of Intel GPUs won't be built.

To determine the model of the Intel GPU, install [pciutils-3.13.0](#) and run `lspci -nn | grep -Ei 'VGA|DISPLAY'` first. It will output some information about the GPU. If the output contains `Meteor Lake`, the GPU model is `MTL`.

Otherwise, follow <https://dgpu-docs.intel.com/devices/hardware-table.html> to find the entry for your GPU and the model of the GPU should be in the "Architecture" column. The Xe and Xe-HPG architectures are covered by the GEN12 model.

Install intel-media by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=$XORG_PREFIX \
-D INSTALL_DRIVER_SYSCONF=OFF \
-D BUILD_TYPE=Release \
-G Ninja \
-W no-dev .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: libigfcmcrt.so

Installed Drivers: iHD_drv_video.so

Installed Directories: /usr/include/igfcmcrt

Short Descriptions

`libigfcmcrt.so` provides API functions which allow running GPU kernels on the render engine

intel-vaapi-driver-2.4.1

Introduction to intel-vaapi-driver

The intel-vaapi-driver package contains a VA API driver for Intel GPUs that are provided with Haswell CPUs and earlier.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/intel/intel-vaapi-driver/releases/download/2.4.1/intel-vaapi-driver-2.4.1.tar.bz2>
- Download MD5 sum: 073fce0f409559109ad2dd0a6531055d
- Download size: 2.8 MB
- Estimated disk space required: 97 MB
- Estimated build time: 0.3 SBU

intel-vaapi-driver Dependencies

Required

[libva-2.22.0](#) and [Xorg build environment](#)

Kernel Configuration

Enable the following options in the kernel configuration. Recompile the kernel if necessary:

```
Device Drivers --->
  Graphics support --->
    <*/M> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) --->
      ...
      [DRM]
    <*/M> Intel 8xx/9xx/G3x/G4x/HD Graphics
          [DRM_I915]
```

Installation of intel-vaapi-driver

Install the driver by running the following commands:

```
./configure $XORG_CONFIG &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Driver: i965_drv_video.so

Liba52-0.8.0

Introduction to Liba52

liba52 is a free library for decoding ATSC A/52 (also known as AC-3) streams. The A/52 standard is used in a variety of applications, including digital television and DVD.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://distfiles.adelielinux.org/source/a52dec/a52dec-0.8.0.tar.gz>
- Download MD5 sum: 4debeed0257f5312e84d92711a5cfcec
- Download size: 448 KB
- Estimated disk space required: 3.7 MB
- Estimated build time: less than 0.1 SBU

Optional

djbfft

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/liba52>

Installation of Liba52

Install liba52 by running the following commands:

```
./configure --prefix=/usr      \
            --mandir=/usr/share/man \
            --enable-shared       \
            --disable-static      \
            CFLAGS="${CFLAGS:--g -O3} -fPIC" &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
cp liba52/a52_internal.h /usr/include/a52dec &&
install -v -m644 -D doc/liba52.txt \
        /usr/share/doc/liba52-0.8.0/liba52.txt
```

Command Explanations

`CFLAGS="${CFLAGS:--g -O3} -fPIC"`: This appends `-fPIC` to `CFLAGS` but use `-g -O3` (the default of this package) instead of an empty string when `CFLAGS` is not set. This is needed to compile liba52 without runtime text relocation. Runtime text relocation is prohibited on x86_64 so `-fPIC` is strictly required. On 32-bit x86 runtime text relocation is allowed but it's insecure and it may waste physical RAM, so `-fPIC` is still better.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`cp liba52/a52_internal.h ...`: Copying this header file into /usr/include/a52dec allows some other programs (such as xine-lib) to compile and link against a system installed liba52.

Contents

Installed Programs: a52dec and extract_a52

Installed Library: liba52.so

Installed Directories: /usr/include/a52dec and /usr/share/doc/liba52-0.8.0

Short Descriptions

a52dec	plays ATSC A/52 audio streams
extract_a52	extracts ATSC A/52 audio from an MPEG stream
liba52.so	provides functions for the programs dealing with ATSC A/52 streams

Libao-1.2.0

Introduction to Libao

The libao package contains a cross-platform audio library. This is useful to output audio on a wide variety of platforms. It currently supports WAV files, OSS (Open Sound System), ESD (Enlighten Sound Daemon), ALSA (Advanced Linux Sound Architecture), NAS (Network Audio system), aRTS (analog Real-Time Synthesizer), and PulseAudio (next generation GNOME sound architecture).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/ao/libao-1.2.0.tar.gz>
- Download MD5 sum: 9f5dd20d7e95fd0dd72df5353829f097
- Download size: 456 KB
- Estimated disk space required: 3.9 MB
- Estimated build time: less than 0.1 SBU

Libao Dependencies

Optional

[a graphical environment](#), [alsa-lib-1.2.12](#), and [PulseAudio-17.0](#)

Installation of Libao

First, fix a gcc-14 issue:

```
sed -i '/limits.h/a #include <time.h>' src/plugins/pulse/ao_pulse.c
```

Install libao by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&  
install -v -m644 README /usr/share/doc/libao-1.2.0
```

Configuring Libao

Config Files

/etc/libao.conf and ~/.libao

Configuration Information

Currently, the only configuration option available is setting the default output device. Issue `man libao.conf` for details.

Contents

Installed Programs: None

Installed Libraries: libao.so and plugins under /usr/lib/ao/plugins-4

Installed Directories: /usr/include/ao, /usr/lib/ao and /usr/share/doc/libao-1.2.0

Short Descriptions

libao.so provides functions for programs wishing to output sound over supported platforms

libaom-3.9.1

Introduction to libaom

The libaom package contains a reference version of the Alliance for Open Media video codec. This codec is a patent free alternative to H.265, and is starting to be used throughout the internet.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://storage.googleapis.com/aom-releases/libaom-3.9.1.tar.gz>
- Download MD5 sum: f31b72e81dd177396e3553576f6515ef
- Download size: 5.3 MB
- Estimated disk space required: 104 MB (add 1.0 GB for tests)
- Estimated build time: 1.0 SBU (with parallelism=4, add 187 SBU for tests)

libaom Dependencies

Recommended

[yasm-1.3.0](#) (or [NASM-2.16.03](#))

Optional

[Doxygen-1.12.0](#)

Note

An Internet connection is needed for some tests of this package.

Installation of libaom

Install libaom by running the following commands:

```
mkdir aom-build &&
cd aom-build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_SHARED_LIBS=1 \
      -D ENABLE_DOCS=no \
      -G Ninja .. &&
ninja
```

To test the results, issue: `ninja runtests`. Note that the tests take an extremely long time to run.

Now, as the `root` user:

```
ninja install &&
rm -v /usr/lib/libaom.a
```

Command Explanations

`-D BUILD_SHARED_LIBS=1`: This switch builds shared versions of the libraries.

`-D ENABLE_DOCS=no`: This switch disables building the documentation because it fails due to an incompatibility with the latest version of [Doxygen-1.12.0](#).

`-D ENABLE_NASM=yes`: Use this switch if you have both [yasm-1.3.0](#) and [NASM-2.16.03](#) installed and wish to use nasm instead of yasm.

Contents

Installed Programs: None

Installed Libraries: libaom.so

Installed Directories: /usr/include/aom

Short Descriptions

`libaom.so` contains functions that provide a reference implementation of the AV1 codec

libass-0.17.3

Introduction to libass

libass is a portable subtitle renderer for the ASS/SSA (Advanced Substation Alpha/Substation Alpha) subtitle format that allows for more advanced subtitles than the conventional SRT and similar formats.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libass/libass/releases/download/0.17.3/libass-0.17.3.tar.xz>
- Download MD5 sum: baed6dfc87bed705c1955cc6b932d7f6
- Download size: 436 KB
- Estimated disk space required: 7.7 MB
- Estimated build time: less than 0.1 SBU

libass Dependencies

Required

[FreeType-2.13.3](#), [FriBidi-1.0.15](#), and [NASM-2.16.03](#)

Recommended

[Fontconfig-2.15.0](#)

Optional

[harfBuzz-9.0.0](#) and [libunibreak](#)

Installation of libass

Install libass by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-fontconfig`: Use this switch if you didn't install Fontconfig.

Contents

Installed Programs: None

Installed Library: libass.so

Installed Directory: /usr/include/ass

Short Descriptions

`libass.so` provides the functions used to render ASS/SSA subtitle format

libcanberra-0.30

Introduction to libcanberra

libcanberra is an implementation of the XDG Sound Theme and Name Specifications, for generating event sounds on free desktops, such as GNOME.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://0pointer.de/lennart/projects/libcanberra/libcanberra-0.30.tar.xz>
- Download MD5 sum: 34cb7e4430afaf6f447c4ebdb9b42072
- Download size: 312 KB
- Estimated disk space required: 7.5 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/libcanberra-0.30-wayland-1.patch>

libcanberra Dependencies

Required

[libvorbis-1.3.7](#)

Recommended

[alsa-lib-1.2.12](#), [gstreamer-1.24.7](#), and [GTK+-3.24.43](#)

Optional

[PulseAudio-17.0](#), [GTK+-2](#) and [tdb](#)

Recommended Sound Theme (Run Time)

[sound-theme-freedesktop-0.8](#), or another theme, for example from [the gnome-look website](#)

Installation of libcanberra

First, apply a patch to fix an issue causing some applications to crash in Wayland-based desktop environments:

```
patch -Np1 -i ../libcanberra-0.30-wayland-1.patch
```

Install libcanberra by running the following commands:

```
./configure --prefix=/usr --disable-oss &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make docdir=/usr/share/doc/libcanberra-0.30 install
```

Command Explanations

`--disable-oss`: disables optional deprecated OSS support

`--disable-gtk3`: disables optional support for GTK+-3

Contents

Installed Programs: canberra-boot and canberra-gtk-play

Installed Libraries: libcanberra-gtk3.so and libcanberra.so

Installed Directories: /usr/lib/libcanberra-0.30, /usr/share/doc/libcanberra-0.30 and /usr/share/gtk-doc/html/libcanberra

Short Descriptions

<code>canberra-gtk-play</code>	is an application used for playing sound events
<code>libcanberra-gtk3.so</code>	contains the libcanberra bindings for GTK+ 3
<code>libcanberra.so</code>	contains the libcanberra API functions

libcddb-1.3.2

Introduction to libcddb

The libcddb is a library that implements the different protocols (CDDBP, HTTP, SMTP) to access data on a CDDB server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/libcddb/libcddb-1.3.2.tar.bz2>
- Download MD5 sum: 8bb4a6f542197e8e9648ae597cd6bc8a
- Download size: 384 KB
- Estimated disk space required: 3.9 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

Note

An Internet connection is needed for some tests of this package.

Installation of libcddb

By default this package accesses `freedb.org`, which is already closed. Modify the default to use `gnudb.gnudb.org` instead, and fix two stale test data files:

```
sed -e '/DEFAULT_SERVER/s/freedb.org/gnudb.gnudb.org/' \  
-e '/DEFAULT_PORT/s/888/&0/' \  
-i include/cddb/cddb_ni.h &&  
sed '/^Genre:/s/Trip-Hop/Electronic/' -i tests/testdata/920ef00b.txt &&  
sed '/DISCID/i# Revision: 42' -i tests/testcache/misc/12340000
```

Fix a problem building with gcc-14:

```
sed -i 's/size_t l;/socklen_t l;/' lib/cddb_net.c
```

Install libcddb by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

To test the results, issue: `make check -k`. The test suite needs the Internet connection. One test fails due to missing test server.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: cddb_query

Installed Library: libcddb.so

Installed Directories: /usr/include/cddb

Short Descriptions

`cddb_query` provides a user interface to a CDDB server

libcdio-2.1.0

Introduction to libcdio

The libcdio is a library for CD-ROM and CD image access. The associated libcdio-cdparanoia library reads audio from the CD-ROM directly as data, with no analog step between, and writes the data to a file or pipe as .wav, .aifc or as raw 16 bit linear PCM.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/libcdio/libcdio-2.1.0.tar.bz2>
- Download MD5 sum: aa7629e8f73662a762f64c444b901055
- Download size: 1.7 MB
- Estimated disk space required: 53 MB (both packages, including checks)
- Estimated build time: 0.2 SBU (using parallelism=4; both packages, including checks)

Additional Downloads

- Required file: <https://ftp.gnu.org/gnu/libcdio/libcdio-paranoia-10.2+2.0.2.tar.bz2>

libcdio Dependencies

Optional

[libcddb-1.3.2](#)

Installation of libcdio

Install libcdio by running the following commands:

```
./configure --prefix=/usr --disable-static &&  
make
```

To test the results, issue: `make check -k`. One test named `realpath` is known to fail.

Now, as the `root` user:

```
make install
```

Now install libcdio-paranoia:

```
tar -xf ../libcdio-paranoia-10.2+2.0.2.tar.bz2 &&
cd libcdio-paranoia-10.2+2.0.2 &&

./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: cdda-player, cd-drive, cd-info, cd-paranoia, cd-read, iso-info, iso-read, and mmc-tool

Installed Library: libcdio.so, libcdio++.so, libcdio_cdda, libcdio_paranoia, libiso9660, libiso9660++, and libudf.so

Installed Directories: /usr/include/cdio and /usr/include/cdio++

Short Descriptions

<code>cd-drive</code>	shows CD-ROM drive characteristics
<code>cd-info</code>	shows information about a CD or CD-image
<code>cd-paranoia</code>	is an audio CD reading utility which includes extra data verification features
<code>cd-read</code>	reads Information from a CD or CD-image
<code>cdda-player</code>	is a simple curses CD player
<code>iso-info</code>	shows Information about an ISO 9660 image
<code>iso-read</code>	reads portions of an ISO 9660 image
<code>mmc-tool</code>	issues libcdio multimedia commands
<code>libcdio.so</code>	contains the primary cdio API functions

libdvdcss-1.4.3

Introduction to libdvdcss

libdvdcss is a simple library designed for accessing DVDs as a block device without having to bother about the decryption.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://get.videolan.org/libdvdcss/1.4.3/libdvdcss-1.4.3.tar.bz2>
- Download MD5 sum: e98239a88af9b2204f9b9d987c2bc71a
- Download size: 380 KB
- Estimated disk space required: 3.3 MB
- Estimated build time: less than 0.1 SBU

libdvdcss Dependencies

Optional (to Create Documentation)

[Doxygen-1.12.0](#)

Installation of libdvdcss

Install libdvdcss by running the following commands:

```
./configure --prefix=/usr \
--disable-static \
```

make

```
--docdir=/usr/share/doc/libdvdcss-1.4.3 &&
```

This package does not come with a test suite.

Now, as the *root* user:

make install

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libdvdcss.so

Installed Directories: /usr/include/dvdcss and /usr/share/doc/libdvdcss-1.4.3

Short Descriptions

libdvdcss.so provides the functionality that is required for transparent DVD access with CSS decryption.

Libdvdread-6.1.3

Introduction to Libdvdread

`libdvdread` is a library which provides a simple foundation for reading DVDs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://get.videolan.org/libdvdread/6.1.3/libdvdread-6.1.3.tar.bz2>
 - Download MD5 sum: 3c58d1624a71a16ff40f55dbaca82523
 - Download size: 388 KB
 - Estimated disk space required: 3.6 MB
 - Estimated build time: less than 0.1 SBU

Installation of Libdvdread

Install libdvdread by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --docdir=/usr/share/doc/libdvdread-6.1.3 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

make install

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libdvdread.so

Installed Directories: /usr/include/dvdread and /usr/share/doc/libdvdread-6.1.3

Short Descriptions

libdvdread.so provides functionality required to access DVDs

Libdvdnav-6.1.1

Introduction to Libdvdnav

libdvdnav is a library that allows easy usage of sophisticated DVD navigation features including DVD menus, multiangle playback, and even interactive DVD games.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://get.videolan.org/libdvdnav/6.1.1/libdvdnav-6.1.1.tar.bz2>
- Download MD5 sum: 46c46cb0294fbd1fcb8a0181818dad15
- Download size: 360 KB
- Estimated disk space required: 3.7 MB
- Estimated build time: less than 0.1 SBU

Libdvdnav Dependencies

Required

[libdvdread-6.1.3](#)

Installation of Libdvdnav

Install libdvdnav by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --docdir=/usr/share/doc/libdvdnav-6.1.1 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libdvdnav.so

Installed Directories: /usr/include/dvdnav and /usr/share/doc/libdvdnav-6.1.1

Short Descriptions

libdvdnav.so is the DVD navigation library

Libdv-1.0.0

Introduction to Libdv

The Quasar DV Codec (libdv) is a software CODEC for DV video, the encoding format used by most digital camcorders. It can be used to copy videos from camcorders using a firewire (IEEE 1394) connection.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/libdv/libdv-1.0.0.tar.gz>
- Download MD5 sum: f895162161cfa4bb4a94c070a7caa6c7
- Download size: 574 KB
- Estimated disk space required: 6.0 MB
- Estimated build time: 0.2 SBU

Libdv Dependencies

Optional

[popt-1.19](#), [sdl12-compat-1.2.68](#), and [a graphical environment](#)

Installation of Libdv

Install libdv by running the following commands:

```
./configure --prefix=/usr \
            --disable-xv \
            --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
install -v -m755 -d      /usr/share/doc/libdv-1.0.0 &&
install -v -m644 README* /usr/share/doc/libdv-1.0.0
```

Command Explanations

`--disable-xv`: This parameter is required if an X Window System is not installed. It also prevents `configure` testing for `libXv` which is only used for an obsolete program `playdv` that will not be built with current linux headers and would also need other obsolete dependencies.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: `dubdv`, `dvconnect`, and `encodeddv`

Installed Library: `libdv.so`

Installed Directories: `/usr/include/libdv` and `/usr/share/doc/libdv-1.0.0`

Short Descriptions

<code>dubdv</code>	inserts audio into a digital video stream
<code>dvconnect</code>	is a small utility to send or capture raw data from and to the camcorder
<code>encodeddv</code>	encodes a series of images to a digital video stream
<code>libdv.so</code>	provides functions for programs interacting with the Quasar DV CODEC

libmad-0.15.1b

Introduction to libmad

libmad is a high-quality MPEG audio decoder capable of 24-bit output.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/mad/libmad-0.15.1b.tar.gz>
- Download MD5 sum: 1be543bc30c56fb6bea1d7bf6a64e66c
- Download size: 491 KB
- Estimated disk space required: 4.2 MB
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/libmad-0.15.1b-fixes-1.patch>

Installation of libmad

Install libmad by running the following commands:

```
patch -Np1 -i ../../libmad-0.15.1b-fixes-1.patch
sed "s@AM_CONFIG_HEADER@AC_CONFIG_HEADERS@g" -i configure.ac &&
touch NEWS AUTHORS ChangeLog &&
autoreconf -fi &&

./configure --prefix=/usr --disable-static &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Some packages check for the `pkg-config` file for libmad. This file is particularly needed so that Cdrdao can recognize the installed libmad.

As the `root` user:

```
cat > /usr/lib/pkgconfig/mad.pc << "EOF"
prefix=/usr
exec_prefix=${prefix}
libdir=${exec_prefix}/lib
includedir=${prefix}/include

Name: mad
Description: MPEG audio decoder
Requires:
Version: 0.15.1b
Libs: -L${libdir} -lmad
Cflags: -I${includedir}
EOF
```

Command Explanations

`touch NEWS AUTHORS ChangeLog`: Prevent autoreconf from returning an error.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libmad.so

Installed Directories: None

Short Descriptions

`libmad.so` is a MPEG audio decoder library

Introduction to libmpeg2

The libmpeg2 package contains a library for decoding MPEG-2 and MPEG-1 video streams. The library is able to decode all MPEG streams that conform to certain restrictions: "constrained parameters" for MPEG-1, and "main profile" for MPEG-2. This is useful for programs and applications needing to decode MPEG-2 and MPEG-1 video streams.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://libmpeg2.sourceforge.net/files/libmpeg2-0.5.1.tar.gz>
- Download MD5 sum: 0f92c7454e58379b4a5a378485bbd8ef
- Download size: 513 KB
- Estimated disk space required: 6 MB
- Estimated build time: 0.1 SBU

libmpeg2 Dependencies

Optional

[a graphical environment](#) and [sdl12-compat-1.2.68](#)

Installation of libmpeg2

Install libmpeg2 by running the following commands:

```
sed -i 's/static const/static/' libmpeg2/idct_mmx.c &&
./configure --prefix=/usr      \
            --enable-shared \
            --disable-static &&
make
```

To test the results, issue: `make check`. To perform a more comprehensive regression test, see the file `test/README` in the source tree.

Now, as the `root` user:

```
make install &&
install -v -m755 -d /usr/share/doc/libmpeg2-0.5.1 &&
install -v -m644 README doc/libmpeg2.txt \
               /usr/share/doc/libmpeg2-0.5.1
```

Command Explanations

`sed -i ...`: This `sed` fixes problems with recent GCC compilers.

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: corrupt_mpeg2, extract_mpeg2 and mpeg2dec

Installed Libraries: libmpeg2.so and libmpeg2convert.so

Installed Directories: /usr/include/mpeg2dec and /usr/share/doc/libmpeg2-0.5.1

Short Descriptions

<code>extract_mpeg2</code>	extracts MPEG video streams from a multiplexed stream
<code>mpeg2dec</code>	decodes MPEG1 and MPEG2 video streams
<code>libmpeg2.so</code>	contains API functions used to decode MPEG video streams
<code>libmpeg2convert.so</code>	contains API functions used for color conversions of MPEG video streams

Introduction to libmusicbrainz

The libmusicbrainz package contains a library which allows you to access the data held on the MusicBrainz server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/metabrainz/libmusicbrainz/releases/download/release-5.1.0/libmusicbrainz-5.1.0.tar.gz>
- Download MD5 sum: 4cc5556aa40ff7ab8f8cb83965535bc3
- Download size: 76 KB
- Estimated disk space required: 6.6 MB (additional 4.4 MB for the API documentation)
- Estimated build time: 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/libmusicbrainz-5.1.0-cmake_fixes-1.patch

libmusicbrainz Dependencies

Required

[CMake-3.30.2](#), [libxml2-2.13.3](#) and [neon-0.33.0](#)

Optional

[Doxygen-1.12.0](#)

Installation of libmusicbrainz

First, fix a problem caused by CMake-3.18.0 and later:

```
patch -Np1 -i ../libmusicbrainz-5.1.0-cmake_fixes-1.patch
```

Fix also a problem caused by libxml-2.12.x:

```
sed -e 's/xmlErrorPtr /const xmlError */' \
    -i src/xmlParser.cc
```

Install libmusicbrainz by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr -D CMAKE_BUILD_TYPE=Release .. &&
make
```

If you have installed [Doxygen-1.12.0](#), optionally build the API documentation:

```
doxygen ..\Doxyfile
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have built the API documentation, install, as the `root` user:

```
rm -rf /usr/share/doc/libmusicbrainz-5.1.0 &&
cp -vr docs/ /usr/share/doc/libmusicbrainz-5.1.0
```

Contents

Installed Programs: None

Installed Library: libmusicbrainz5.so

Installed Directory: /usr/include/libmusicbrainz5 and /usr/share/doc/libmusicbrainz-5.1.0

Short Descriptions

libmusicbrainz5.so contains API functions for accessing the MusicBrainz database

libogg-1.3.5

Introduction to libogg

The libogg package contains the Ogg file structure. This is useful for creating (encoding) or playing (decoding) a single physical bit stream.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/ogg/libogg-1.3.5.tar.xz>
- Download MD5 sum: 3178c98341559657a15b185bf5d700a5
- Download size: 420 KB
- Estimated disk space required: 3.5 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

Installation of libogg

Install libogg by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --docdir=/usr/share/doc/libogg-1.3.5 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libogg.so

Installed Directories: /usr/include/ogg and /usr/share/doc/libogg-1.3.5

Short Descriptions

libogg.so provides the functions required for programs to read or write Ogg formatted bit streams

libplacebo-7.349.0

Introduction to libplacebo

The libplacebo package contains a library for processing image and video primitives and shaders. It also includes a high quality rendering pipeline that supports OpenGL and Vulkan.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/haasn/libplacebo/archive/v7.349.0/libplacebo-7.349.0.tar.gz>
- Download MD5 sum: 8dd72edf2ec5f1918770a317ef8107a5
- Download size: 828 KB
- Estimated disk space required: 36 MB
- Estimated build time: 0.1 SBU (With tests, both using parallelism=4)

libplacebo Dependencies

Required

[FFmpeg-7.0.2](#) and [Glad-2.0.6](#)

Recommended

[Glslang-14.3.0](#) and [Vulkan-Loader-1.3.294](#)

Optional

[Little CMS-2.16](#) [libleunwind-1.6.2](#), [dovi_tool](#), [Nuklear](#), and [xxHash](#)

Installation of libplacebo

Install libplacebo by running the following commands:

```
mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D tests=true \
    -D demos=false \
&&
ninja
```

To test the results, issue: `ninja test`. One test, `opengl_surfaceless.c`, is known to fail.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D demos=false`: This switch disables building the demo programs because building plplay is currently broken.

`-D tests=true`: This switch enables building the code necessary to run the tests.

Contents

Installed Programs: None

Installed Libraries: libplacebo.so

Installed Directories: /usr/include/libplacebo

Short Descriptions

<code>libplacebo.so</code>	processes image and video primitives and shaders and provides a high quality rendering pipeline for OpenGL and Vulkan
----------------------------	---

libsamplerate-0.2.2

Introduction to libsamplerate

`libsamplerate` is a sample rate converter for audio.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libsndfile/libsamplerate/releases/download/0.2.2/libsamplerate-0.2.2.tar.xz>
- Download MD5 sum: 97c010fc25156c33cddc272c1935afab
- Download size: 3.2 MB
- Estimated disk space required: 15 MB (add 2 MB for tests)
- Estimated build time: less than 0.1 SBU (add 0.3 SBU for tests)

libsamplerate Dependencies

Optional

[alsa-lib-1.2.12](#), [libsndfile-1.2.2](#), and [fftw-3.3.10](#) (for tests)

Installation of libsamplerate

Install `libsamplerate` by running the following commands:

```
./configure --prefix=/usr      \
            --disable-static \
            --docdir=/usr/share/doc/libsamplerate-0.2.2 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Program: None

Installed Library: `libsamplerate.so`

Installed Directory: `/usr/share/doc/libsamplerate-0.2.2`

libsndfile-1.2.2

Introduction to libsndfile

`Lbsndfile` is a library of C routines for reading and writing files containing sampled audio data.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libsndfile/libsndfile/releases/download/1.2.2/libsndfile-1.2.2.tar.xz>
- Download MD5 sum: 04e2e6f726da7c5dc87f8cf72f250d04
- Download size: 716 KB
- Estimated disk space required: 12 MB (add 10 MB for tests)
- Estimated build time: 0.3 SBU (add 0.3 SBU for tests)

libsndfile Dependencies

Recommended

[FLAC-1.4.3](#), [Opus-1.5.2](#), and [libvorbis-1.3.7](#)

Optional

[alsa-lib-1.2.12](#), [LAME-3.100](#), [mpg123-1.32.7](#), [Speex-1.2.1](#), and [SQLite-3.46.1](#)

Installation of libsndfile

Install libsndfile by running the following commands:

```
./configure --prefix=/usr \
            --docdir=/usr/share/doc/libsndfile-1.2.2 &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: `sndfile-cmp`, `sndfile-concat`, `sndfile-convert`, `sndfile-deinterleave`, `sndfile-info`, `sndfile-interleave`, `sndfile-metadata-get`, `sndfile-metadata-set`, `sndfile-play`, and `sndfile-salvage`

Installed Library: `libsndfile.so`

Installed Directory: `/usr/share/doc/libsndfile-1.2.2`

Short Descriptions

<code>sndfile-cmp</code>	compares two audio files
<code>sndfile-concat</code>	concatenates two or more audio files
<code>sndfile-convert</code>	converts a sound file from one format to another
<code>sndfile-deinterleave</code>	splits a multi-channel into multiple single channel files
<code>sndfile-info</code>	displays information about a sound file
<code>sndfile-interleave</code>	converts multiple single channel files into a multi-channel file
<code>sndfile-metadata-get</code>	retrieves metadata from a sound file
<code>sndfile-metadata-set</code>	sets metadata in a sound file
<code>sndfile-play</code>	plays a sound file
<code>sndfile-salvage</code>	salvages the audio data from WAV files which are more than 4 GB in size
<code>libsndfile.so</code>	contains the <code>libsndfile</code> API functions

libva-2.22.0

Introduction to libva

The libva package contains a library which provides access to hardware accelerated video processing, using hardware to accelerate video processing in order to offload the central processing unit (CPU) to decode and encode compressed digital video. The VA API video decode/encode interface is platform and window system independent targeted at Direct Rendering Infrastructure (DRI) in the X Window System however it can potentially also be used with direct framebuffer and graphics sub-systems for video output. Accelerated processing includes support for video decoding, video encoding, subpicture blending, and rendering.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/intel/libva/archive/2.22.0/libva-2.22.0.tar.gz>
- Download MD5 sum: 8dba13313d724b5e930f06e65b7437e2
- Download size: 296 KB
- Estimated disk space required: 3.7 MB
- Estimated build time: less than 0.1 SBU

libva Dependencies

Required

[Xorg build environment](#) and [libdrm-2.4.122](#)

Recommended

[Mesa-24.1.5](#)

Recommended (Runtime)

The VA API driver suitable for the hardware in your system: [intel-vaapi-driver-2.4.1](#) (for Intel GPUs provided with Haswell CPUs or earlier), [intel-media-24.2.5](#) (for Intel GPUs provided with Broadwell CPUs or later), and [Mesa-24.1.5](#) (providing the r600, radeonsi, and nouveau VA API drivers, for the ATI/AMD Radeon HD 2xxx GPUs and later, and [supported NVIDIA GPUs](#); there is a circular dependency, read the Mesa page for information on how to break it)

Optional

[Doxygen-1.12.0](#), [Wayland-1.23.0](#), and [intel-gpu-tools](#)

Installation of libva

Install libva by running the following commands:

```
cd build &&
meson setup --prefix=$XORG_PREFIX --buildtype=release &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Contents

Installed Programs: None

Installed Libraries: libva-drm.so, libva-glx.so, libva.so, libva-wayland.so, and libva-x11.so

Installed Directory: \$XORG_PREFIX/include/va

Short Descriptions

`libva.so` contains API functions which provide access to hardware accelerated video processing

libvdpau-1.5

Introduction to libvdpau

The libvdpau package contains a library which implements the VDPAU library.

VDPAU (Video Decode and Presentation API for Unix) is an open source library (libvdpau) and API originally designed by Nvidia for its GeForce 8 series and later GPU hardware targeted at the X Window System. This VDPAU API allows video programs to offload portions of the video decoding process and video post-processing to the GPU video-hardware.

Currently, the portions capable of being offloaded by VDPAU onto the GPU are motion compensation (mo comp), inverse discrete cosine transform (iDCT), VLD (variable-length decoding) and deblocking for MPEG-1, MPEG-2, MPEG-4 ASP (MPEG-4 Part 2), H.264/MPEG-4 AVC and VC-1, WMV3/WMV9 encoded videos. Which specific codecs of these that can be offloaded to the GPU depends on the version of the GPU hardware; specifically, to also decode MPEG-4 ASP (MPEG-4 Part 2), Xvid/OpenDivX (DivX 4), and DivX 5 formats, a GeForce 200M (2xxM) Series (the eleventh generation of Nvidia's GeForce graphics processing units) or newer GPU hardware is required.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/vdpau/libvdpau/-/archive/1.5/libvdpau-1.5.tar.bz2>
- Download MD5 sum: 148a192110e7a49d62c0bf9ef916c099

- Download size: 140 KB
- Estimated disk space required: 4.6 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

libvdpau Dependencies

Required

[Xorg Libraries](#)

Recommended (Runtime)

The VDPAU driver suitable for the hardware in your system: [libvdpau-va-gl-0.4.2](#) (for Intel GPUs) and [Mesa-24.1.5](#) (providing the `r600`, `radeonsi`, and `nouveau` VDPAU drivers, for the ATI/AMD Radeon HD 2xxx GPUs and later, and [supported NVIDIA GPUs](#); Mesa must be built after this package for these drivers)

Optional

[Doxygen-1.12.0](#), [Graphviz-12.1.0](#), and [texlive-20240312](#) or [install-tl-unx](#)

Installation of libvdpau

Install libvdpau by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=$XORG_PREFIX .. &&
ninja
```

To test the results, issue: `ninja test`. There is only one test for this package, `dlclose`, and it is known to fail on some systems.

Now, as the `root` user:

```
ninja install
```

If doxygen is present at build time place the documentation in a versioned directory as the `root` user:

```
[ -e $XORG_PREFIX/share/doc/libvdpau ] && mv -v $XORG_PREFIX/share/doc/libvdpau{,1.5}
```

Contents

Installed Programs: None

Installed Library: libvdpau.so

Installed Directories: \$XORG_PREFIX/{include,lib}/vdpau

Short Descriptions

`libvdpau.so` contains functions to offload portions of the video decoding process and video post-processing to the GPU video-hardware

libvdpau-va-gl-0.4.2

Introduction to libvdpau-va-gl

The libvdpau-va-gl package contains a library which implements the VDPAU library. Libvdpau_va_gl uses OpenGL under the hood to accelerate drawing and scaling and the VA-API (if available) to accelerate video decoding. For now VA-API is available on some Intel chips, and on some AMD video adapters with the help of the libvdpau driver.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Libvdpau-va-gl Driver Download (HTTP): <https://github.com/i-rinat/libvdpau-va-gl/archive/v0.4.2/libvdpau-va-gl-0.4.2.tar.gz>

- Libvdpau-va-gl Driver Download MD5 sum: 8db21dcfd5cd14c6ec51b992e20369dc
- Libvdpau-va-gl Driver Download size: 120 KB
- Estimated disk space required: 4.9 MB
- Estimated build time: less than 0.1 SBU (add 1.1 SBU for tests)

libvdpau-va-gl Dependencies

Required

[CMake-3.30.2](#), [libvdpau-1.5](#), [libva-2.22.0](#), and [Mesa-24.1.5](#)

Optional

[Doxygen-1.12.0](#), [Graphviz-12.1.0](#), and [texlive-20240312](#) or [install-tl-unx](#)

Installation of libvdpau-va-gl

Install libvdpau-va-gl by running the following commands:

```
mkdir build &&
cd      build &&

cmake -D CMAKE_BUILD_TYPE=Release -D CMAKE_INSTALL_PREFIX=$XORG_PREFIX ... &&
make
```

To test the results, issue: `make check`. The tests must be run from an Xorg environment.

Now, as the `root` user:

```
make install
```

Configuration

To allow libvdpau to find libvdpau-va-gl, set an environment variable. As the `root` user:

```
echo "export VDPAU_DRIVER=va_gl" >> /etc/profile.d/xorg.sh
```

Contents

Installed Programs: None

Installed Library: libvdpau_va_gl.so

Installed Directories: None

Short Descriptions

libvdpau_va_gl.so	contains functions to implement the OpenGL backend to the VDPAU (Video Decode and Presentation API for Unix) API
-------------------	--

libvorbis-1.3.7

Introduction to libvorbis

The libvorbis package contains a general purpose audio and music encoding format. This is useful for creating (encoding) and playing (decoding) sound in an open (patent free) format.

This package is known to build and work properly using LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/vorbis/libvorbis-1.3.7.tar.xz>
- Download MD5 sum: 50902641d358135f06a8392e61c9ac77
- Download size: 1.1 MB
- Estimated disk space required: 15 MB

- Estimated build time: 0.1 SBU

libvorbis Dependencies

Required

[libogg-1.3.5](#)

Optional

[Doxxygen-1.12.0](#) and [texlive-20240312](#) (or [install-tl-unx](#)) (specifically, pdflatex and htlatex) to build the PDF documentation

Installation of libvorbis

Install libvorbis by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

To test the results, issue: `make -j1 check`.

Now, as the `root` user:

```
make install &&
install -v -m644 doc/Vorbis* /usr/share/doc/libvorbis-1.3.7
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-docs`: This switch enables building the documentation in formats other than the supplied html.

Contents

Installed Programs: None

Installed Libraries: libvorbis.so, libvorbisenc.so and libvorbisfile.so

Installed Directories: /usr/include/vorbis and /usr/share/doc/libvorbis-1.3.7

Short Descriptions

`libvorbis.so` provides the functions used to read and write sound files

libvpx-1.14.1

Introduction to libvpx

This package, from the WebM project, provides the reference implementations of the VP8 Codec (used in most current HTML5 video) and of the next-generation VP9 Codec.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/webmproject/libvpx/archive/v1.14.1/libvpx-1.14.1.tar.gz>
- Download MD5 sum: 1d2a1c594587ee1f26a4f017becddcd5
- Download size: 5.4 MB
- Estimated disk space required: 68 MB (add 1.6 GB for tests)
- Estimated build time: 0.5 SBU (Using parallelism=4; add approx 11 SBU for tests using parallelism=4, including download time)

libvpx Dependencies

Recommended

[yasm-1.3.0](#) or [NASM-2.16.03](#), and [Which-2.21](#) (so `configure` can find yasm)

Optional

[cURL-8.9.1](#) (to download test files) and [Doxygen-1.12.0](#) (to build documentation)

Note

An Internet connection is needed for some tests of this package.

Installation of libvpx

Install libvpx by running the following commands:

```
sed -i 's/cp -p/cp/' build/make/Makefile &&  
  
mkdir libvpx-build &&  
cd libvpx-build &&  
  
./configure --prefix=/usr \
            --enable-shared \
            --disable-static &&  
make
```

To test the results, issue: `LD_LIBRARY_PATH=. make test`. The test suite downloads many files as part of its test process. A few parts of it will use all available cores.

Now, as the `root` user:

```
make install
```

Command Explanations

`sed ...`: This command corrects the ownership and permissions of installed files.

`mkdir libvpx-build && cd libvpx-build`: The libvpx developers recommend building in a dedicated build directory.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--disable-vp8`: This switch prevents building support for the VP8 codec.

`--disable-vp9`: This switch prevents building support for the VP9 codec.

`--target=generic-gnu`: This switch disables optimizations specific to x86 and x86-64, allowing to build this package without `nasm` and `yasm` installed.

`LD_LIBRARY_PATH=.`: This is needed for the test suite to use the library that was just built.

Contents

Installed Programs: vpxdec and vpxenc

Installed Libraries: libvpx.so

Installed Directories: /usr/include/vpx

Short Descriptions

<code>vpxdec</code>	is the WebM Project VP8 and VP9 decoder
<code>vpxenc</code>	is the WebM project VP8 and VP9 encoder
<code>libvpx.so</code>	provides functions to use the VP8 and VP9 video codecs

mlt-7.26.0

Introduction to mlt

The mlt package is the Media Lovin Toolkit. It is an open source multimedia framework, designed and developed for television broadcasting. It provides a toolkit for broadcasters, video editors, media players, transcoders, web streamers and many more types of applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mltframework/mlt/releases/download/v7.26.0/mlt-7.26.0.tar.gz>
- Download MD5 sum: 14b822bc386db5c0fd2c9e5116c04d73
- Download size: 1.6 MB
- Estimated disk space required: 29 MB
- Estimated build time: 0.1 SBU (Using parallelism=4)

mlt Dependencies

Required

[frei0r-plugins-1.8.0](#) and [Qt-6.7.2](#)

Optional

[Doxygen-1.12.0](#), [fftw-3.3.10](#), [libexif-0.6.24](#), [SDL2-2.30.6](#), [qt5-components-5.15.14](#), [JACK](#), [MOVEit](#), [SoX](#), and [vid.stab](#)

Installation of mlt

Install mlt by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D MOD_QT=OFF \
      -D MOD_QT6=ON \
      -W no-dev .. &&
make
```

This package does not come with a test suite. However a test .mp4 file can be played in a local graphical environment with `./out/bin/melt <filename>.mp4`.

Note

This application uses advanced graphical capabilities. In some cases, firmware for your specific graphics adaptor may be needed. See [the section called "Firmware for Video Cards"](#) for more information.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: melt-7 and melt (symlink to melt-7)

Installed Libraries: libmlt-7.so, libmlt++-7.so, and over twenty plugins

Installed Directories: /usr/include/mlt-7, /usr/lib/mlt-7, /usr/lib/cmake/MLT7, and /usr/share/mlt-7

Short Descriptions

`melt` is a test tool for mlt

Opus-1.5.2

Introduction to Opus

Opus is a lossy audio compression format developed by the Internet Engineering Task Force (IETF) that is particularly suitable for interactive speech and audio transmission over the Internet. This package provides the Opus development library and headers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/opus/opus-1.5.2.tar.gz>
- Download MD5 sum: c40b3a1fbdbb9a7aa178600b88200c76
- Download size: 7.4 MB
- Estimated disk space required: 33 MB (with tests)
- Estimated build time: 0.5 SBU (with tests)

Optional

[Doxygen-1.12.0](#) and [texlive-20240312](#) (or [install-tl-unx](#))

Installation of Opus

Install Opus by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D docdir=/usr/share/doc/opus-1.5.2 &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

Contents

Installed Programs: None

Installed Library: libopus.so

Installed Directory: /usr/include/opus and /usr/share/doc/opus-1.5.2

Short Descriptions

`libopus.so` provides the functions used to read and write Opus format

Pipewire-1.2.3

Introduction to Pipewire

The pipewire package contains a server and userspace API to handle multimedia pipelines. This includes a universal API to connect to multimedia devices, as well as sharing multimedia files between applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/pipewire/pipewire/-/archive/1.2.3/pipewire-1.2.3.tar.bz2>
- Download MD5 sum: 9a37ae78752ac7462bb0660f61a4a870

- Download size: 1.7 MB
- Estimated disk space required: 69 MB (with tests)
- Estimated build time: 0.6 SBU (with tests, both using parallelism=4)

Pipewire Dependencies

Recommended

[BlueZ-5.77](#), [gstreamer-1.24.7](#), [gst-plugins-base-1.24.7](#), [PulseAudio-17.0](#), [SBC-2.0](#), [v4l-utils-1.28.1](#), and [Wireplumber-0.5.5](#) (runtime)

Optional

[alsa-lib-1.2.12](#), [Avahi-0.8](#), [fdk-aac-2.0.3](#), [FFmpeg-7.0.2](#), [libcanberra-0.30](#), [libdrm-2.4.122](#) (for one example and libcamera support), [libxcb-1.17.0](#), [libsndfile-1.2.2](#), [libusb-1.0.27](#), [Opus-1.5.2](#), [SDL2-2.30.6](#) (for some examples), [Valgrind-3.23.0](#), [Vulkan-Loader-1.3.294](#), [Xorg Libraries](#), [Doxygen-1.12.0](#) and [Graphviz-12.1.0](#) (for documentation), [JACK](#), [IdacBT](#), [libcamera](#), [libmysofa](#), and [xmltoman](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/pipewire>

Installation of Pipewire

Install pipewire by running the following commands:

```
mkdir build &&
cd      build &&

meson setup ..           \
    --prefix=/usr          \
    --buildtype=release     \
    -D session-managers="[]" &&
ninja
```

To test the result, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D session-managers="[]"`: This switch allows specifying the session managers to build as subprojects. We specify an empty list to prevent `meson` from downloading any external copy of the session managers. A Pipewire session manager recommended by the PipeWire developers for general use, [Wireplumber-0.5.5](#), is available as a separate BLFS package. Build it after installing Pipewire if you need to run the Pipewire daemon (for example in order to support [Snapshot-46.3](#)).

`-D docs=true`: This switch enables the generation of HTML documentation. The optional dependencies for documentation need to be installed for this to work.

`-D man=true`: This switch enables the generation of manual pages. The optional dependencies for documentation need to be installed for this to work.

`-D ffmpeg=enabled`: This switch enables using ffmpeg for audio conversion as a SPA backend.

`-D vulkan=enabled`: This switch enables using Vulkan for video rendering as a SPA backend.

Contents

Installed Programs: pipewire, pw-cat, pw-cli, pw-config, pw-dot, pw-dump, pw-jack, pw-link, pw-loopback, pw-metadata, pw-mididump, pw-mon, pw-profiler, pw-reserve, pw-top, pw-v4l2, spa-acp-tool, spa-inspect, spa-json-dump, spa-monitor, spa-resample, and pipewire-aes67, pipewire-avb, pipewire-pulse (symlinks to pipewire), pw-dsdplay, pw-encplay, pw-midiplay, pw-midirecord, pw-play, pw-record (symlinks to pw-cat)

Installed Libraries: libpipewire-0.3.so and 60 modules below /usr/lib/pipewire-0.3 and /usr/lib/spa-0.2

Installed Directories: /usr/include/pipewire-0.3, /usr/include/spa-0.2, /usr/lib/alsa-lib, /usr/lib/pipewire-0.3, /usr/lib/spa-0.2, /usr/share/alsa-card-profile, /usr/share/pipewire, and /usr/share/spa-0.2

Short Descriptions

<code>pipewire</code>	is a service that allows access to multimedia devices and allows sharing of multimedia files between applications
<code>pipewire-pulse</code>	starts a Pulseaudio-compatible version of the pipewire service
<code>pw-cat</code>	allows you to play or record media using a pipewire instance
<code>pw-cli</code>	allows you to interact with a pipewire instance
<code>pw-config</code>	allows you to interact with pipewire's configuration system for debugging purposes or syntax checking
<code>pw-dot</code>	lists all of the daemons and objects in use by pipewire
<code>pw-dump</code>	dumps debug messages from a local or remote pipewire instance to the console
<code>pw-jack</code>	runs JACK applications on a pipewire instance
<code>pw-link</code>	links pipewire ports together
<code>pw-loopback</code>	initializes a loopback link between two different pipewire ports. This is useful for testing and debugging
<code>pw-metadata</code>	inspects, adds, and removes metadata for objects
<code>pw-mididump</code>	dumps MIDI messages from a file to the screen
<code>pw-mon</code>	allows you to monitor pipewire instances
<code>pw-profiler</code>	tracks memory usage and API calls used by pipewire
<code>pw-reserve</code>	reserves or monitors a device via D-Bus
<code>pw-top</code>	displays real-time performance information from pipewire
<code>pw-v4l2</code>	runs v4l2 applications on a pipewire instance
<code>spa-acp-tool</code>	inspects the card profile of a given sound card
<code>spa-inspect</code>	allows you to inspect pipewire plugins
<code>spa-json-dump</code>	dumps the current pipewire configuration in JSON format
<code>spa-monitor</code>	allows you to monitor pipewire plugins
<code>spa-resample</code>	resamples a given file
<code>libpipewire-0.3.so</code>	contains API functions for handling multimedia pipelines

PulseAudio-17.0

Introduction to PulseAudio

PulseAudio is a sound system for POSIX OSes, meaning that it is a proxy for sound applications. It allows you to do advanced operations on your sound data as it passes between your application and your hardware. Operations such as transferring the audio to a different machine, changing the sample format or channel count, and mixing several sounds into one are easily achieved using a sound server.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/pulseaudio/releases/pulseaudio-17.0.tar.xz>
- Download MD5 sum: c4a3596a26ff4b9dc0c394dd1d4f8ee
- Download size: 1.5 MB
- Estimated disk space required: 42 MB (with tests)
- Estimated build time: 0.2 SBU (Using parallelism=4; add 0.2 SBU for tests)

PulseAudio Dependencies

Required

[libsndfile-1.2.2](#)

Recommended

[alsa-lib-1.2.12](#), [dbus-1.14.10](#), [GLib-2.80.4](#), [Speex-1.2.1](#) and [Xorg Libraries](#)

Optional

[Avahi-0.8](#), [BlueZ-5.77](#), [Doxygen-1.12.0](#) (for documentation), [fftw-3.3.10](#), [GTK+-3.24.43](#), [libsamplerate-0.2.2](#), [SBC-2.0](#) (Bluetooth support), [Valgrind-3.23.0](#), [JACK](#), [libasynchs](#), [LIRC](#), [ORC](#), [soxr](#), [TDB](#), and [WebRTC AudioProcessing](#)

Installation of PulseAudio

Install PulseAudio by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D database=gdbm   \
            -D doxygen=false  \
            -D bluez5=disabled \
            ..
            &&
ninja
```

To test the results, issue: `ninja test`. One test fails if the tests are not run as the root user, but this can be ignored.

Now, as the `root` user:

```
ninja install
```

Running PulseAudio as a system-wide daemon is possible but not recommended. See <https://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/SystemWide/> for more information. While still as the `root` user, remove the D-Bus configuration file for the system wide daemon to avoid creating unnecessary system users and groups:

```
rm /usr/share/dbus-1/system.d/pulseaudio-system.conf
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D bluez5=disabled`: This switch prevents a runtime error if [dbus-1.14.10](#) and [SBC-2.0](#) are both installed but [BlueZ-5.77](#) is not installed. Remove this if you have installed all three packages.

`-D doxygen=false`: This allows the package to compile if [Doxygen-1.12.0](#) is not installed. Remove this if you have installed this and wish to build the documentation.

Configuring PulseAudio

Config Files

There are system wide configuration files: `/etc/pulse/daemon.conf`, `/etc/pulse/client.conf`, `/etc/pulse/default.pa`, and user configuration files with the same names in `~/.config/pulse`. User configuration files take precedence over system wide ones.

Configuration Information

You may have to configure the audio system. You can start pulseaudio in command line mode using `pulseaudio -c` and then list various information and change settings. See `man pulse-cli-syntax`.

If pulseaudio was working but you no-longer have sound, after checking for hardware issues (speakers or headphones not connected, external amplifier not connected) you may need to fix it.

One suggestion is to close the application, such as firefox where sound has stopped working, then run: `pacctl list short sinks` followed by: `pacmd set-default-sink <sink #>` and then restart the application.

If that does not work, a more drastic approach often works. After closing the application, close pulseaudio, either using `pulseaudio --kill` or, if that fails, `killall -KILL pulseaudio` and then `rm -rf ~/.config/pulse/*` (and perhaps also `rm -rf ~/.pulse/*` if you have used a very old version of pulse on this machine), then run `pulseaudio --verbose` to restart it. If the daemon starts, restart the application. See `man pulseaudio` for more options.

Contents

Installed Programs: pacat, pacmd, pactl, padsp, pamon (symlink to pacat), paplay (symlink to pacat), parec (symlink to pacat), parecord (symlink to pacat), qpaeq, pasuspender, pax11publish, pulseaudio, and start-

pulseaudio-x11

Installed Libraries: libpulse.so, libpulse-mainloop-glib.so, libpulse-simple.so, libpulsecommon-17.0.so, libpulsecore-17.0.so, and libpulsedsp.so

Installed Directories: /etc/pulse, /usr/include/pulse, /usr/lib/cmake/PulseAudio, /usr/lib/{pulseaudio,pulse-17.0}, /usr/libexec/pulse, and /usr/share/pulseaudio

Short Descriptions

<code>pacat</code>	plays back or records raw or encoded audio streams on a PulseAudio sound server
<code>pacmd</code>	is a tool used to reconfigure a PulseAudio sound server during runtime
<code>pactl</code>	is used to control a running PulseAudio sound server
<code>padsp</code>	is the PulseAudio OSS Wrapper
<code>pamon</code>	is a symbolic link to <code>pacat</code>
<code>paplay</code>	is used to play audio files on a PulseAudio sound server
<code>parec</code>	is a symbolic link to <code>pacat</code>
<code>parecord</code>	is a symbolic link to <code>pacat</code>
<code>pasuspender</code>	is a tool that can be used to tell a local PulseAudio sound server to temporarily suspend access to the audio devices, which allows other applications to access them directly
<code>pax11publish</code>	is the PulseAudio X11 Credential Utility
<code>pa-info</code>	is a shell script that prints various information relating to the Pulseaudio server currently in use
<code>pulseaudio</code>	is a networked low-latency sound server for Linux
<code>qpaeq</code>	is an equalizer interface for PulseAudio equalizer sinks (Requires fftw-3.3.10 at build time)
<code>start-pulseaudio-x11</code>	starts PulseAudio and registers it to the X11 session manager

SBC-2.0

Introduction to SBC

The SBC package is a digital audio encoder and decoder used to transfer data to Bluetooth audio output devices like headphones or loudspeakers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.kernel.org/pub/linux/bluetooth/sbc-2.0.tar.xz>
- Download MD5 sum: 5613357181daeffd71e971c6f8470f8d
- Download size: 268 KB
- Estimated disk space required: 2.8 MB
- Estimated build time: less than 0.1 SBU

SBC Dependencies

Optional

[libsndfile-1.2.2](#)

Installation of SBC

Install SBC by running the following commands:

```
./configure --prefix=/usr --disable-static --disable-tester &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

--disable-tester: This disables the SBC tester. Remove it if you have installed [libsndfile-1.2.2](#).

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: sbcdec, sbcenc, and sbcinfo

Installed Library: libsbc.so

Installed Directory: /usr/include/sbc

Short Descriptions

sbcdec	is a SBC decoder utility
sbcenc	is a SBC encoder utility
sbcinfo	is a subband codec (SBC) analyzer
libsbc.so	contains the SBC API functions

sdl12-compat-1.2.68

Introduction to sdl12-compat

The Simple DirectMedia Layer (SDL for short) is a cross-platform library designed to make it easy to write multimedia software, such as games and emulators. This code is a compatibility layer; it provides a binary and source compatible API for programs written against SDL 1.2, but it uses SDL 2.0 behind the scenes.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/libsdl-org/sdl12-compat/archive/release-1.2.68/sdl12-compat-release-1.2.68.tar.gz>
- Download MD5 sum: 04ed17494dad2131b37e0300f8af31ec
- Download size: 452 KB
- Estimated disk space required: 7.1 MB
- Estimated build time: less than 0.1 SBU

SDL Dependencies

Required

[CMake-3.30.2](#), [GLU-9.0.3](#), and [SDL2-2.30.6](#)

Installation of sdl12-compat

Install sdl12-compat by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=RELEASE \
      .. &&
make
```

Now, as the `root` user:

```
make install &&
rm -vf /usr/lib/libSDLmain.a
```

Testing SDL

If you wish, test the sdlcompat package using the included test programs. It is not required to install any of the resulting binaries to validate the installation.

You'll need to manually run all the test programs (they are listed in the `README` file in this directory). Many of them will need to be manually killed, and you'll need to turn your speakers on with the volume at a suitable level.

Contents

Installed Program: `sdl-config`

Installed Libraries: `libSDL.so`

Installed Directories: `/usr/include/SDL`

Short Descriptions

`sdl-config` determines the compile and linker flags that should be used to compile and link programs that use `libSDL`

`libSDL.so` contains functions that provide compatibility links to SDL2 low level functions for audio, keyboard, mouse, joystick, 3D hardware via OpenGL, and the 2D frame buffer across multiple platforms

SDL2-2.30.6

Introduction to SDL2

The Simple DirectMedia Layer Version 2 (SDL2 for short) is a cross-platform library designed to make it easy to write multimedia software, such as games and emulators.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.libsdl.org/release/SDL2-2.30.6.tar.gz>
- Download MD5 sum: ab12cc1cf58a5dd25e69c924acb93402
- Download size: 7.2 MB
- Estimated disk space required: 186 MB (with docs)
- Estimated build time: 0.6 SBU (using parallelism=4; with docs)

SDL2 Dependencies

Recommended

[libxkbcommon-1.7.0](#), [wayland-protocols-1.36](#), and [Xorg Libraries](#) (if those are not present, the corresponding modules are not built)

Optional

[alsa-lib-1.2.12](#), [Doxygen-1.12.0](#) (to create documentation), [ibus-1.5.30](#), [libsamplerate-0.2.2](#), [libunwind-1.6.2](#), [NASM-2.16.03](#), [pipewire-1.2.3](#), [PulseAudio-17.0](#), [a graphical environment](#), [DirectFB](#), [fcitx](#), [jack](#), and [sndio](#)

Installation of SDL2

Install SDL2 by running the following commands:

```
./configure --prefix=/usr &&  
make
```

If you have [Doxygen-1.12.0](#) installed and want to build the html documentation, run the following commands:

```
pushd docs &&  
doxygen &&  
popd
```

Note

If you wish to build and run the package regression tests, do not delete the static libraries below until after the tests are built.

Now, as the `root` user:

```
make install      &&
rm -v /usr/lib/libSDL2*.a
```

If you built the documentation, install it as the `root` user:

```
install -v -m755 -d      /usr/share/doc/SDL2-2.30.6/html &&
cp -Rv docs/output/html/* /usr/share/doc/SDL2-2.30.6/html
```

Testing SDL2

If you wish to, test the installation of SDL2 using the included test programs. None of the resulting binaries need to be installed. Issue the following commands to build the test programs:

```
cd test &&
./configure &&
make
```

Each of the test programs (they are listed in the `README` file in this directory) will need to be run individually. Many of them will need to be manually killed. Additionally, speakers need to be on with the volume at a suitable level.

Command Explanations

`rm -v /usr/lib/libSDL2*.a`: Normally static libraries can be disabled with a `--disable-static` option to configure, but that breaks the build in this package.

Contents

Installed Program: `sdl2-config`

Installed Libraries: `libSDL2.so`

Installed Directories: `/usr/include/SDL2`, `/usr/lib/cmake/SDL2`, and `/usr/share/doc/SDL2-2.30.6`

Short Descriptions

`sdl2-config` determines the compile and linker flags that should be used to compile and link programs that use `libSDL2`

`libSDL2.so` contains functions that provide low level access to audio, keyboard, mouse, joystick, 3D hardware via OpenGL, and the 2D frame buffer across multiple platforms

sound-theme-freedesktop-0.8

Introduction to Sound Theme Freedesktop

The Sound Theme Freedesktop package contains sound themes for the desktop.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://people.freedesktop.org/~mccann/dist/sound-theme-freedesktop-0.8.tar.bz2>
- Download MD5 sum: d7387912cf275282d1ec94483cb2f62
- Download size: 472 KB
- Estimated disk space required: 2 MB
- Estimated build time: less than 0.1 SBU

Installation of Sound Theme Freedesktop

Install Sound Theme Freedesktop by running the following commands:

```
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Directories: /usr/share/sounds/freedesktop

SoundTouch-2.3.3

Introduction to SoundTouch

The SoundTouch package contains an open-source audio processing library that allows changing the sound tempo, pitch and playback rate parameters independently from each other.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.surina.net/soundtouch/soundtouch-2.3.3.tar.gz>
- Download MD5 sum: 489c1050315965f69a9c0ad949744d73
- Download size: 596 KB
- Estimated disk space required: 7.6 MB
- Estimated build time: 0.2 SBU

Installation of SoundTouch

Note

The tarball expands to 'soundtouch' instead of the expected 'soundtouch-2.3.3'

Install SoundTouch by running the following commands:

Caution

The `bootstrap` command below fails if the `ACLOCAL` environment variable is set as specified in [Xorg-7](#). If it is used, `ACLOCAL` needs to be unset for this package and then reset for other packages.

```
./bootstrap &&
./configure --prefix=/usr \
            --docdir=/usr/share/doc/soundtouch-2.3.3 &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-openmp`: This switch adds support for running the algorithms in parallel across several processor cores using OpenMP implementation provided by GCC.

Contents

Installed Program: soundstretch

Installed Library: libSoundTouch.so

Installed Directories: /usr/include/soundtouch and /usr/share/doc/soundtouch-2.3.3

Short Descriptions

soundstretch is a program for processing WAV audio files by modifying sound tempo, pitch and playback rate properties independently from each other

libSoundTouch.so contains SoundTouch API functions

Speex-1.2.1

Introduction to Speex

Speex is an audio compression format designed specifically for speech. It is well-adapted to internet applications and provides useful features that are not present in most other CODECs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/speex/speex-1.2.1.tar.gz>
- Download MD5 sum: fe7bf610883ff202092b92c72fe0fe3e
- Download size: 1020 KB
- Estimated disk space required: 7.8 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Download (HTTP): <https://downloads.xiph.org/releases/speex/speexdsp-1.2.1.tar.gz>
- Download MD5 sum: e6eb5ddef743a362c8018f260b91dca5
- Download size: 904 KB
- Estimated disk space required: 5.5 MB
- Estimated build time: less than 0.1 SBU

Speex Dependencies

Required

[libogg-1.3.5](#)

Optional

[Valgrind-3.23.0](#)

Installation of Speex

This package consists of two separate tarballs. They need to be extracted and built independently.

Install Speex by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --docdir=/usr/share/doc/speex-1.2.1 &&
make
```

This package does not come with a test suite.

As the `root` user:

```
make install
```

Now extract and install the speexdsp package:

```
cd ..          &&
tar -xf speexdsp-1.2.1.tar.gz &&
cd speexdsp-1.2.1          &&

./configure --prefix=/usr      \
            --disable-static \
            --docdir=/usr/share/doc/speexdsp-1.2.1 &&
make
```

Again, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: speexenc and speexdec

Installed Libraries: libspeex.so and libspeexdsp.so

Installed Directories: /usr/include/speex and /usr/share/doc/speex-1.2.1

Short Descriptions

<code>speexdec</code>	decodes a Speex file and produces a WAV or raw file
<code>speexenc</code>	encodes a WAV or raw file using Speex
<code>libspeex.so</code>	provides functions for the audio encoding/decoding programs
<code>libspeexdsp.so</code>	is a speech processing library that goes along with the Speex codec

Taglib-2.0.1

Introduction to Taglib

Taglib is a library used for reading, writing and manipulating audio file tags and is used by applications such as Amarok and VLC.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://taglib.org/releases/taglib-2.0.1.tar.gz>
- Download MD5 sum: e1f2ef858bddf65eb17e43043c3da10b
- Download size: 1.4 MB
- Estimated disk space required: 16 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

Taglib Dependencies

Required

[CMake-3.30.2](#) and [utfcpp-4.0.5](#)

Optional (for tests)

[Cppunit](#)

Installation of Taglib

Install Taglib by running the following commands:

```

mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -D CMAKE_BUILD_TYPE=Release \
      -D BUILD_SHARED_LIBS=ON \
      .. &&
make

```

The test suite requires Cppunit but it's not a BLFS package.

Now, as the `root` user:

```
make install
```

Contents

- Installed Programs:** taglib-config
- Installed Libraries:** libtag.so and libtag_c.so
- Installed Directories:** /usr/include/taglib

Short Descriptions

`taglib-config` is a tool used to print information about the taglib installation

v4l-utils-1.28.1

Introduction to v4l-utils

v4l-utils provides a series of utilities for media devices, allowing the ability to handle the proprietary formats available from most webcams (libv4l), and providing tools to test V4L devices.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.linuxtv.org/downloads/v4l-utils/v4l-utils-1.28.1.tar.xz>
- Download MD5 sum: 6716de513a1fd2e1edb404a46a455855
- Download size: 1.2 MB
- Estimated disk space required: 34 MB
- Estimated build time: 0.4 SBU (using parallelism=4)

v4l-utils Dependencies

Recommended

[alsa-lib-1.2.12](#), [GLU-9.0.3](#), and [libjpeg-turbo-3.0.1](#)

Optional

[Doxygen-1.12.0](#), [Qt-6.7.2](#) (for qv4l2 and qvidcap), [SDL2-2.30.6](#), both [LLVM-18.1.7](#) (with Clang and target BPF) and [libbpf](#) (for infrared remote control decoders based on BPF), and [SDL_image](#)

Installation of v4l-utils

Note

If you've installed both LLVM and libbpf (not in BLFS), either enable the BPF target (via `-D LLVM_TARGETS_TO_BUILD=`) when building LLVM, or disable the infrared remote control decoders based on BPF:

```
sed -i '/^ir_bpf_enabled/s/=.*=/ false/' utils/keytable/meson.build
```

Install v4l-utils by running the following commands:

```

mkdir build &&
cd build &&

meson setup .. \
    --prefix=/usr \
    --buildtype=release \
    -D gconv=disabled \
    -D doxygen-doc=disabled &&
ninja

```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

If you also wish to install the contrib programs, add:

```

for prog in v4l2gl v4l2grab
do
    cp -v contrib/test/$prog /usr/bin
done

```

Command Explanations

`-D gconv=disabled`: This switch disables the erroneous installation of gconv-modules that would break glibc.

Contents

Installed Programs: cec-compliance, cec-ctl, cec-follower, cx18-ctl, decode_tm6000, dvb-fe-tool, dvb-format-convert, dvbv5-daemon, dvbv5-scan, dvbv5-zap, ir-ctl, ir-keytable, ivtv-ctl, media-ctl, qv4l2, qvidcap, rds-ctl, v4l2-compliance, v4l2-ctl, v4l2-dbg, v4l2gl, v4l2grab, and v4l2-sysfs-path

Installed Library: libdvbv5.so, libv4l1.so, libv4l2.so, libv4l2rds.so, libv4lconvert.so, v4l1compat.so, and v4l2convert.so

Installed Directories: /etc/rc_keymaps, /lib/udev/rc_keymaps, /usr/include/libdvbv5, and /usr/lib/libv4l

Short Descriptions

<code>cx18-ctl</code>	is a tool to handle cx18 based devices
<code>decode_tm6000</code>	decodes multiplexed formats from TM5600/TM6000 USB devices
<code>dvb-fe-tool</code>	is a command line tool for digital TV services
<code>dvb-format-convert</code>	is a tool meant to convert among different file formats. It is compliant with version 5 of the DVB API
<code>dvbv5-scan</code>	is a command line frequency scanning tool for digital TV services that are compliant with version 5 of the DVB API
<code>dvbv5-zap</code>	is a command line tuning tool for digital TV services that is compliant with version 5 of the DVB API
<code>ir-keytable</code>	is a tool that lists the Remote Controller devices, allows one to get/set IR keycode/scancode tables, test events generated by IR, and to adjust other Remote Controller options
<code>ivtv-ctl</code>	is a utility that can control many card settings, like changing the inputs from tuner to svideo or composite, setting video resolution or changing video mode (PAL, SECAM, NTSC)
<code>media-ctl</code>	is a utility used to configure V4L2 devices
<code>qv4l2</code>	is used to test video4linux capture devices
<code>qvidcap</code>	is used to capture video from a V4L2 device
<code>rds-ctl</code>	is a utility for decoding raw RDS data from V4L2 Radio devices and offers simple ways to access the received RDS information
<code>v4l2-compliance</code>	is a compliance test tool
<code>v4l2-ctl</code>	is a tool to control v4l2 controls from the cmdline
<code>v4l2-dbg</code>	is a tool to directly get and set registers of v4l2 devices
<code>v4l2gl</code>	captures images using libv4l and stores them as PPM files
<code>v4l2grab</code>	captures images using libv4l and stores them as PPM files, while also allowing for additional options such as setting the amount of frames captured and saving the file as a RAW file
<code>v4l2-sysfs-path</code>	checks the media devices installed on a machine and the corresponding device nodes

Wireplumber-0.5.5

Introduction to Wireplumber

The Wireplumber package contains a session and policy manager for Pipewire.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://gitlab.freedesktop.org/pipewire/wireplumber/-/archive/0.5.5/wireplumber-0.5.5.tar.bz2>
- Download MD5 sum: 258234d08926fd1f6824fb4b134720f9
- Download size: 404 KB
- Estimated disk space required: 82 MB (with tests)
- Estimated build time: 0.3 SBU (with tests)

Wireplumber Dependencies

Required

[GLib-2.80.4](#), [pipewire-1.2.3](#), and [Systemd-256.4](#) (runtime, rebuilt with [Linux-PAM-1.6.1](#))

Recommended

[Lua-5.4.7](#)

Optional

[Doxygen-1.12.0](#), [lxml-5.3.0](#), [sphinx-8.0.2](#), [sphinx_rtd_theme-2.0.0](#), and [Breathe](#)

Installation of Wireplumber

Install Wireplumber by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release -D system-lua=true .. &&
ninja
```

To test the results, issue: `ninja test`.

Now, as the `root` user:

```
ninja install
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`-D system-lua=true`: This switch enables using the system version of Lua. Omit this switch if you have not installed Lua, but note that meson will download and install its own version.

Configuring Wireplumber

Configuration Information

In order for Wireplumber to be used by Pipewire, Wireplumber must be configured to start when a desktop environment is started.

To configure Wireplumber to start when a desktop environment is started, run the following commands as the `root` user to enable the `systemd` user units:

```
systemctl enable --global pipewire.socket &&
systemctl enable --global wireplumber
```

Now that Wireplumber is configured to start when a desktop environment is started, it is recommended that you log out of your session and log back in again.

Contents

Installed Programs: wireplumber, wpctl, and wpexec

Installed Libraries: libwireplumber-0.5.so

Installed Directories: /usr/include/wireplumber-0.5, /usr/lib/wireplumber-0.5, and /usr/share/wireplumber

Short Descriptions

wireplumber	is a session and policy manager for Pipewire
wpctl	controls and queries information from Wireplumber
wpexec	executes Wireplumber scripts, which are often written in Lua
libwireplumber-0.5.so	contains functions that allow other programs to control Wireplumber

x264-20240812

Introduction to x264

x264 package provides a library for encoding video streams into the H.264/MPEG-4 AVC format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/x264-20240812.tar.xz>
- Download MD5 sum: 515794f289b34f59f16553f2438867ed
- Download size: 724 KB
- Estimated disk space required: 15 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

x264 Dependencies

Recommended

[NASM-2.16.03](#)

Optional

[ffms2](#), [gpac](#) or [liblsmash](#)

Installation of x264

Install x264 by running the following commands:

```
./configure --prefix=/usr \
            --enable-shared \
            --disable-cli &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-cli`: This switch disables building the command-line encoder which is redundant since it requires FFmpeg for most of the input formats.

--disable-asm: Use this switch if you didn't install NASM.

Contents

Installed Programs: None

Installed Library: libx264.so

Installed Directory: None

Short Descriptions

libx264.so provides the functions used to encode video streams into the H.264/MPEG-4 AVC format

x265-3.6

Introduction to x265

x265 package provides a library for encoding video streams into the H.265/HEVC format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): https://bitbucket.org/multicoreware/x265_git/downloads/x265_3.6.tar.gz
- Download MD5 sum: 99997ecc8ee4d3575ba7715c759ad3bb
- Download size: 1.6 MB
- Estimated disk space required: 45 MB
- Estimated build time: 0.5 SBU (using parallelism=4)

x265 Dependencies

Required

[CMake-3.30.2](#)

Recommended

[NASM-2.16.03](#)

Optional

[numactl](#)

Installation of x265

Install x265 by running the following commands:

```
mkdir bld &&
cd bld &&

cmake -D CMAKE_INSTALL_PREFIX=/usr \
      -W no-dev ../source &&
make
```

This package does not come with a test suite.

Important

If downgrading from the bleeding-edge x265-20240812 (it had been mistakenly used in the BLFS development book for several days), remove the shared library from x265-20240812 as the `root` user:

```
rm -fv /usr/lib/libx265.so.212
```

This is for preventing `ldconfig` from recreating the symlink `libx265.so` with the target `/usr/lib/libx265.so.212` which seems to be "a newer version" of `libx265.so.209` from `x265-3.6`.

To install the package, first remove any old library versions. After installation, remove a static library. As the `root` user:

```
make install &&
rm -vf /usr/lib/libx265.a
```

Command Explanations

`-W no-dev`: This switch is used to suppress warnings intended for the package developers.

`rm -vf /usr/lib/libx265.a`: BLFS does not recommend using static libraries.

Contents

Installed Program: x265

Installed Library: libx265.so

Installed Directories: None

Short Descriptions

`libx265.so` provides the functions used to encode video streams into the H.265/HEVC format

xine-lib-1.2.13

Introduction to Xine Libraries

The Xine Libraries package contains xine libraries. These are useful for interfacing with external plug-ins that allow the flow of information from the source to the audio and video hardware.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/xine/xine-lib-1.2.13.tar.xz>
- Download MD5 sum: 9e1be39857b7a3cd7cc0f2b96331ff22
- Download size: 4.8 MB
- Estimated disk space required: 98 MB (with API documentation)
- Estimated build time: 0.5 SBU (Using parallelism=4; with API documentation)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/xine-lib-1.2.13-upstream_fixes-1.patch

Xine Libraries Dependencies

Required

a graphical environment, [FFmpeg-7.0.2](#), and at least one of: [alsa-lib-1.2.12](#), [PulseAudio-17.0](#), or [JACK](#)

Recommended

[libdvdnav-6.1.1](#)

Optional

[AAlib-1.4rc5](#), [FAAD2-2.11.1](#), [FLAC-1.4.3](#), [gdk-pixbuf-2.42.12](#), [GLU-9.0.3](#), [ImageMagick-7.1.1-36](#), [liba52-0.8.0](#), [libdvdcss-1.4.3](#), [libmad-0.15.1b](#), [libmng-2.0.3](#), [libnsl-2.0.1](#), [libva-2.22.0](#) (requires [GLU-9.0.3](#)), [libvdpau-1.5](#), [libvorbis-1.3.7](#), [libvpx-1.14.1](#), [Mesa-24.1.5](#), [Samba-4.20.4](#), [sdl12-compat-1.2.68](#), [Speex-1.2.1](#), [Doxygen-1.12.0](#) (to create the API documentation), [y4l-utils-1.28.1](#), [DirectFB](#), [libbluray](#), [libcaca](#), [libdca](#), [libFAME](#), [libmodplug](#), [libtheora](#), [musepack](#), [VCDImager](#), and [WavPack](#),

Installation of Xine Libraries

First, fix building this package with ffmpeg-7.0 and later:

```
patch -Np1 -i ../../xine-lib-1.2.13-upstream_fixes-1.patch
```

Install Xine Libraries by running the following commands:

```
./configure --prefix=/usr \
--disable-vcd \
--with-external-dvdnav \
--docdir=/usr/share/doc/xine-lib-1.2.13 &&
make
```

To create the API documentation, Doxygen must be installed and issue the following command:

```
doxygen doc/Doxyfile
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Note

When installing, the Makefile does some additional linking. If you do not have Xorg in /usr, the LIBRARY_PATH variable needs to be defined for the root user. If using sudo to assume root, use the -E option to pass your current environment variables for the install process.

If you built the API documentation, issue the following commands as the `root` user to install it:

```
install -v -m755 -d /usr/share/doc/xine-lib-1.2.13/api &&
install -v -m644 doc/api/* \
/usr/share/doc/xine-lib-1.2.13/api
```

Command Explanations

`--disable-vcd`: This switch is required to compile Xine Lib without [VCDImager](#) installed. Remove this option if you have installed VCDImager.

`--with-external-dvdnav`: This switch is required, because internal libraries cannot decrypt DVD.

`--docdir=/usr/share/doc/xine-lib-1.2.13`: This switch causes the documentation to be installed into a versioned directory instead of the default `/usr/share/doc/xine-lib`.

`--disable-vaapi`: use this switch if [libva-2.22.0](#) is installed and [GLU-9.0.3](#) is not, to prevent a build failure.

Contents

Installed Programs: xine-config and xine-list-1.2

Installed Libraries: libxine.so and numerous plugin modules and video extensions under /usr/lib/xine/plugins/2.8

Installed Fonts: Output display engine fonts located in /usr/share/xine-lib/fonts

Installed Directories: /usr/include/xine, /usr/lib/xine, /usr/share/xine-lib, and /usr/share/doc/xine-lib-1.2.13

Short Descriptions

<code>xine-config</code>	provides information to programs trying to link with the xine libraries
<code>xine-list-1.2</code>	is used to get supported file type information from xine-lib
<code>libxine.so</code>	provides the API for processing audio/video files

XviD-1.3.7

Introduction to XviD

XviD is an MPEG-4 compliant video CODEC.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xvid.com/downloads/xvidcore-1.3.7.tar.gz>
- Download MD5 sum: 5c6c19324608ac491485dbb27d4da517
- Download size: 804 KB
- Estimated disk space required: 7.1 MB
- Estimated build time: 0.1 SBU

XviD Dependencies

Optional

[yasm-1.3.0](#) or [NASM-2.16.03](#)

Installation of XviD

Note

This package tarball expands to `xvidcore`, not the expected `xvidcore-1.3.7`.

Install XviD by running the following commands:

```
cd build/generic &&
sed -i 's/^LN_S=@LN_S@/& -f -v/' platform.inc.in &&
./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
sed -i '/libdir.*STATIC_LIB/ s/^/#/' Makefile &&
make install &&

chmod -v 755 /usr/lib/libxvidcore.so.4.3 &&
install -v -m755 -d /usr/share/doc/xvidcore-1.3.7/examples &&
install -v -m644 ../../doc/* /usr/share/doc/xvidcore-1.3.7 &&
install -v -m644 ../../examples/* \
    /usr/share/doc/xvidcore-1.3.7/examples
```

Command Explanations

`sed -i 's/^LN_S=@LN_S@/& -f -v/' platform.inc.in`: Fix error during `make install` if reinstalling or upgrading.

`sed -i '/libdir.*STATIC_LIB/ s/^/#/' Makefile`: This command disables installing the static library.

Contents

Installed Programs: None

Installed Library: libxvidcore.so

Installed Directory: /usr/share/doc/xvidcore-1.3.7

Short Descriptions

`libxvidcore.so` provides functions to encode and decode most MPEG-4 video data

This chapter contains programs involved with audio file manipulation; that is to say playing, recording, ripping and the other common things which people want to do. It also includes a package used to render text to speech using your system's audio hardware. To use much of this software, you will need to have the kernel sound drivers installed.

Audacious-4.4

Introduction to Audacious

Audacious is an audio player.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://distfiles.audacious-media-player.org/audacious-4.4.tar.bz2>
- Download MD5 sum: 812367242529db8ca0fb0fdb71ffc91f
- Download size: 624 KB
- Estimated disk space required: 14 MB (with GTK support)
- Estimated build time: 0.2 SBU (with GTK support)

Additional Downloads

Required Plugins

- Download (HTTP): <https://distfiles.audacious-media-player.org/audacious-plugins-4.4.tar.bz2>
- Download MD5 sum: 0946f32277afc60ec12510bd09507016
- Download size: 1.7 MB
- Estimated disk space required: 40 MB (with GTK support)
- Estimated build time: 2.0 SBU (with GTK support)

Audacious Dependencies

Required

[GTK+-3.24.43](#) and [Qt-6.7.2](#)

Recommended

[libarchive-3.7.4](#)

Optional

[Valgrind-3.23.0](#)

Recommended (for Plugins)

[mpg123-1.32.7](#), [FFmpeg-7.0.2](#), and [neon-0.33.0](#) (for online mp3 and ogg radio)

Optional (for Plugins)

[alsa-lib-1.2.12](#), [cURL-8.9.1](#), [FAAD2-2.11.1](#), [FLAC-1.4.3](#), [LAME-3.100](#), [libxml2-2.13.3](#), [libcdio-2.1.0](#) (to identify and play CDs), [libnotify-0.8.3](#), [libsndfile-1.2.2](#), [libvorbis-1.3.7](#), [pipewire-1.2.3](#), [PulseAudio-17.0](#), [SDL2-2.30.6](#), [ampache-browser](#), [adplug](#), [The Bauer stereophonic-to-binaural DSP \(bs2b\) library](#), [FluidSynth](#), [JACK](#) (requires [libsamplerate-0.2.2](#)), [libcue](#), [libmodplug](#), [libmms](#), [libopenmpt](#), [libsidplayfp](#), [LIRC](#), [sndio](#), [SoX](#), and [WavPack](#)

Installation of Audacious

Installation of the Main Audacious Program

Install Audacious by running the following commands (you may wish to change the buildstamp to another string):

```
mkdir build &&
cd      build &&
meson setup --prefix=/usr      \
```

```
--buildtype=release \
-D gtk=true \
-D qt=true \
-D buildstamp=BLFS \
-D libarchive=true \
.. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Installation of Audacious Plugins

Install the required plugins package by issuing the following commands:

```
tar -xf ../../audacious-plugins-4.4.tar.bz2 &&
cd audacious-plugins-4.4 &&

mkdir build &&
cd build &&

meson setup \
--prefix=/usr \
--buildtype=release \
-D gtk=true \
-D qt=true \
.. &&
ninja
```

Now, as the `root` user:

```
ninja install
```

Command Explanations

`-D buildstamp=BLFS`: This switch appends the given text to the version string.

`-D gtk=true`: This option enables GTK support for the graphical user interface.

`-D qt=true`: This option enables Qt support for the graphical user interface.

`-D libarchive=true`: This adds libarchive support, for reading compressed module sets or skins.

`-D valgrind=true`: The option enables Valgrind analysis support.

Contents

Installed Programs: `audacious` and `audtool`

Installed Libraries: `libaudcore.so`, `libaudgui.so`, `libaudqt.so`, `libaudtag.so`, and several plugin libraries under `/usr/lib/audacious/` sub-directories

Installed Directories: `/usr/include/{audacious,libaudcore,libaudgui,libaudqt}`, `/usr/lib/audacious`, and `/usr/share/audacious`

Short Descriptions

`audacious` is an audio player and is a descendant of XMMS

`audtool` is a small tool to modify the behavior of a running `audacious` instance

CDParanoia-III-10.2

Introduction to CDParanoia

The CDParanoia package contains a CD audio extraction tool. This is useful for extracting `.wav` files from audio CDs. A CDDA capable CDROM drive is needed. Practically all drives supported by Linux can be used.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/cdparanoia/cdparanoia-III-10.2.src.tgz>
- Download MD5 sum: b304bbe8ab63373924a744eac9ebc652
- Download size: 179 KB
- Estimated disk space required: 2.9 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/cdparanoia-III-10.2-gcc_fixes-1.patch

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/cdparanoia>

Installation of CDParanoia

Note

This package does not support parallel build.

Install CDParanoia by running the following commands:

```
patch -Np1 -i ../../cdparanoia-III-10.2-gcc_fixes-1.patch &&
./configure --prefix=/usr --mandir=/usr/share/man &&
make -j1
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&
chmod -v 755 /usr/lib/libcdda_* .so .0 .10 .2 &&
rm -fv /usr/lib/libcdda_* .a
```

Contents

Installed Program: cdparanoia

Installed Libraries: libcdda_interface.so and libcdda_paranoia.so

Installed Directories: None

Short Descriptions

<code>cdparanoia</code>	is used for 'ripping' an audio-cd. Ripping is the process of digitally extracting music from an audio CD
<code>libcdda_interface</code>	contains functions used by <code>cdparanoia</code> , as well as other packages, which can automatically identify if a CD device is CDDA compatible
<code>libcdda_paranoia</code>	contains functions used by <code>cdparanoia</code> , as well as other packages, which provide data verification, synchronization, error handling and scratch reconstruction capability

LAME-3.100

Introduction to LAME

The LAME package contains an MP3 encoder and optionally, an MP3 frame analyzer. This is useful for creating and analyzing compressed audio files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/lame/lame-3.100.tar.gz>

- Download MD5 sum: 83e260acbe4389b54fe08e0bdbf7cddb
- Download size: 1.5 MB
- Estimated disk space required: 9.7 MB
- Estimated build time: 0.1 SBU

LAME Dependencies

Optional

[Dmalloc](#), [Electric Fence](#), [libsndfile-1.2.2](#) and [NASM-2.16.03](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/lame>

Installation of LAME

Prevent the source code directory from being mistakenly hardcoded as a shared library search path in the installed programs:

```
sed -i -e 's/^(\s*hardcode_libdir_flag_spec\s*=)\s*/\1/' configure
```

Install LAME by running the following commands:

```
./configure --prefix=/usr --enable-mp3rtp --disable-static &&
make
```

To test the results, issue: `LD_LIBRARY_PATH=libmp3lame/.libs make test`.

Now, as the `root` user:

```
make pkghtmldir=/usr/share/doc/lame-3.100 install
```

Command Explanations

`--enable-mp3rtp`: This switch enables building the encode-to-RTP program.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-nasm`: This option enables the use of [NASM-2.16.03](#) to compile optimized assembly routines for 32-bit x86. Note that this has no effect on x86_64.

Contents

Installed Programs: lame and mp3rtp

Installed Library: libmp3lame.so

Installed Directories: /usr/include/lame and /usr/share/doc/lame-3.100

Short Descriptions

<code>lame</code>	creates MP3 audio files from raw PCM or .wav data
<code>mp3rtp</code>	is used to encode MP3 with RTP streaming of the output
<code>libmp3lame.so</code>	provides the functions necessary to convert raw PCM and WAV files to MP3 files

mpg123-1.32.7

Introduction to mpg123

The mpg123 package contains a console-based MP3 player. It claims to be the fastest MP3 decoder for Unix.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/mpg123/mpg123-1.32.7.tar.bz2>
- Download MD5 sum: 45820aa078659c98c42bb676025365db

- Download size: 1.1 MB
- Estimated disk space required: 16 MB (with tests)
- Estimated build time: 0.2 SBU (with tests)

mpg123 Dependencies

Recommended

[alsa-lib-1.2.12](#)

Optional

[PulseAudio-17.0](#), [sdl12-compat-1.2.68](#), [JACK](#), [OpenAL](#) and [PortAudio](#)

Installation of mpg123

Install mpg123 by running the following commands:

```
./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Contents

Installed Programs: mpg123, mpg123-id3dump, mpg123-strip, and out123

Installed Libraries: libmpg123.so, libout123.so, and libsyn123.so

Installed Directory: /usr/lib/mpg123

Short Descriptions

mpg123	is used for playing MP3 files via the console
mpg123-id3dump	is a tool to dump ID3 meta data from MPEG audio files using libmpg123
mpg123-strip	extracts only MPEG frames from a stream using libmpg123 (stdin to stdout)
out123	plays raw PCM audio to an output device
libmpg123.so	contains the mpg123 API functions
libout123.so	contains the out123 API functions
libsyn123.so	contains some audio signal synthesis and format conversion functions

pavucontrol-6.1

Introduction to pavucontrol

PulseAudio Volume Control (pavucontrol) is a simple GTK based volume control tool ("mixer") for the PulseAudio sound server. In contrast to classic mixer tools, this one allows you to control both the volume of hardware devices and of each playback stream separately.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.freedesktop.org/software/pulseaudio/pavucontrol/pavucontrol-6.1.tar.xz>
- Download MD5 sum: 51743b9bc9eb01959bf3c770facc6555
- Download size: 168 KB
- Estimated disk space required: 5.2 MB
- Estimated build time: 0.2 SBU

pavucontrol Dependencies

Required

[Gtkmm-4.14.0](#), [JSON-GLib-1.8.0](#), [libsigc++-3.6.0](#), and [PulseAudio-17.0](#)

Optional

[libcanberra-0.30](#) and [Lynx-2.9.2](#)

Installation of pavucontrol

Install pavucontrol by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr --buildtype=release -D lynx=false .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install &&
mv /usr/share/doc/pavucontrol /usr/share/doc/pavucontrol-6.1
```

Command Explanations

`-D lynx=false`: This switch disables generating the README file in text format. Remove this switch if you want the README file in text format and have [Lynx-2.9.2](#) installed.

Contents

Installed Program: pavucontrol

Installed Libraries: None

Installed Directory: /usr/share/pavucontrol and /usr/share/doc/pavucontrol-6.1

Short Descriptions

`pavucontrol` is a GUI configuration tool for sound settings using pulsaudio

pnmixer-0.7.2

Introduction to Pnmixer

The Pnmixer package provides a lightweight volume control with a tray icon.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/nicklan/pnmixer/releases/download/v0.7.2/pnmixer-v0.7.2.tar.gz>
- Download MD5 sum: e9f17f56c50de39393030a96e343427b
- Download size: 236 KB
- Estimated disk space required: 2.5 MB
- Estimated build time: less than 0.1 SBU

Pnmixer Dependencies

Required

[alsa-utils-1.2.12](#), [GTK+-3.24.43](#), and [libnotify-0.8.3](#)

Installation of Pnmixer

Install Pnmixer by running the following commands:

```
mkdir build &&
cd build &&

cmake -D CMAKE_INSTALL_PREFIX=/usr .. &&
make
```

This package does not have a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: pnmixer

Installed Libraries: None

Installed Directories: /usr/share/pnmixer

Short Descriptions

`pnmixer` is a lightweight volume control that sits in a tray

vorbis-tools-1.4.2

Introduction to Vorbis Tools

The Vorbis Tools package contains command-line tools useful for encoding, playing or editing files using the Ogg codec.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.xiph.org/releases/vorbis/vorbis-tools-1.4.2.tar.gz>
- Download MD5 sum: 998fca293bd4e4bdc2b96fb70f952f4e
- Download size: 1.3 MB
- Estimated disk space required: 12 MB
- Estimated build time: 0.1 SBU

Vorbis Tools Dependencies

Required

[libvorbis-1.3.7](#)

Optional (required to build the `ogg123` program)

[libao-1.2.0](#)

Optional

[cURL-8.9.1](#), [FLAC-1.4.3](#), [Speex-1.2.1](#), and [libkate](#)

Installation of Vorbis Tools

First, apply a sed to fix a gcc-14 issue:

```
sed -i '/i18n.h/i #include "utf8.h"' ogginfo/codec_skeleton.c
```

Next, apply a sed to fix a security vulnerability in 'oggenc':

```
sed -i 's/start+1,/start + strspn(start, PATH_SEPS),/' oggenc/platform.c
```

Install Vorbis Tools by running the following commands:

```
./configure --prefix=/usr --enable-vcut &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--enable-vcut`: This parameter is used so that the `vcut` program is built as it is not by default.

Configuring Vorbis Tools

Config Files

`/etc/libao.conf`, `~/.libao` and `~/.ogg123rc`

Configuration Information

Issue `man libao.conf` for information about setting the default output device. Also see `/usr/share/doc/vorbis-tools-1.4.2/ogg123rc-example`.

Contents

Installed Programs: `ogg123`, `oggdec`, `oggenc`, `ogginfo`, `vcut`, and `vorbiscomment`

Installed Libraries: None

Installed Directory: `/usr/share/doc/vorbis-tools-1.4.2`

Short Descriptions

<code>ogg123</code>	is a command-line audio player for Ogg Vorbis streams
<code>oggdec</code>	is a simple decoder which converts Ogg Vorbis files into PCM audio files (WAV or raw)
<code>oggenc</code>	is an encoder that turns raw, WAV or AIFF files into an Ogg Vorbis stream
<code>ogginfo</code>	prints information stored in an audio file
<code>vcut</code>	splits a file into two files at a designated cut point
<code>vorbiscomment</code>	is an editor that changes information in the audio file metadata tags

Chapter 44. Video Utilities

This chapter always seems to be the favorite chapter. It's probably because there is a lot of satisfaction in playing your first video when you have spent so much time getting to that point. All those libraries, all the configurations and your reward is that you finally get to watch a movie. Not to worry though, there is always one more CODEC to install.

FFmpeg-7.0.2

Introduction to FFmpeg

FFmpeg is a solution to record, convert and stream audio and video. It is a very fast video and audio converter and it can also acquire from a live audio/video source. Designed to be intuitive, the command-line interface (`ffmpeg`) tries to figure out all the parameters, when possible. FFmpeg can also convert from any sample rate to any other, and resize video on the fly with a high quality polyphase filter. FFmpeg can use a Video4Linux compatible video source and any Open Sound System audio source.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ffmpeg.org/releases/ffmpeg-7.0.2.tar.xz>
- Download MD5 sum: e22725fc3738e314d71a7fb32f2336db
- Download size: 10 MB

- Estimated disk space required: 228 MB (add 1.5 GB for the FATE suite/tests, add 792 MB for documentation)
- Estimated build time: 1.1 SBU (using parallelism=4; add 3.2 SBU (with THREADS=4) to run the FATE suite after sample files are downloaded; add 0.3 SBU for doxygen docs)

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/ffmpeg-7.0.2-chromium_method-1.patch

FFmpeg Dependencies

Recommended

[libaom-3.9.1](#), [libass-0.17.3](#), [fdk-aac-2.0.3](#), [FreeType-2.13.3](#), [LAME-3.100](#), [libvorbis-1.3.7](#), [libvpx-1.14.1](#), [Opus-1.5.2](#), [x264-20240812](#), [x265-3.6](#), and [NASM-2.16.03](#) or [yasm-1.3.0](#)

Recommended for desktop use

[alsa-lib-1.2.12](#), [libva-2.22.0](#), [SDL2-2.30.6](#)

Recommended for systems with Intel Video

[libvdpau-1.5](#) and [libvdpau-va-gl-0.4.2](#)

Optional

[Doxygen-1.12.0](#), [Fontconfig-2.15.0](#), [FriBidi-1.0.15](#), [frei0r-plugins-1.8.0](#), [libcdio-2.1.0](#) (to identify and play CDs), [libdrm-2.4.122](#) (for “kmsggrab” input), [libjxl-0.10.3](#), [libwebp-1.4.0](#), [opencv-4.10.0](#), [OpenJPEG-2.5.2](#), [GnuTLS-3.8.7.1](#), [PulseAudio-17.0](#), [Samba-4.20.4](#), [Speex-1.2.1](#), [texlive-20240312](#) (or [install-tl-unx](#)) for PDF and PS documentation, [v4l-utils-1.28.1](#), [Vulkan-Loader-1.3.294](#), [XviD-1.3.7](#), [a graphical environment](#), [dav1d](#), [Flite](#), [GSM](#), [libaacplus](#), [libbluray](#), [libcaca](#), [libcelt](#), [libdc1394](#), [libdca](#), [libiec61883](#), [libilbc](#), [libmodplug](#), [libnut](#) (Git checkout), [librtmp](#), [libssh](#), [libtheora](#), [OpenAL](#), [OpenCore AMR](#), [Srt](#), [Schroedinger](#), [TwoLAME](#), [vo-aaenc](#), [vo-amrwbenc](#), and [ZVBI](#)

Installation of FFmpeg

First, apply a patch that adds an API necessary for some packages to build:

```
patch -Np1 -i .../ffmpeg-7.0.2-chromium_method-1.patch
```

Install FFmpeg by running the following commands:

```
./configure --prefix=/usr \
    --enable-gpl \
    --enable-version3 \
    --enable-nonfree \
    --disable-static \
    --enable-shared \
    --disable-debug \
    --enable-libaom \
    --enable-libass \
    --enable-libfdk-aac \
    --enable-libfreetype \
    --enable-libmp3lame \
    --enable-libopus \
    --enable-libvorbis \
    --enable-libvpx \
    --enable-libx264 \
    --enable-libx265 \
    --enable-openssl \
    --ignore-tests=enforced-flv-av1 \
    --docdir=/usr/share/doc/ffmpeg-7.0.2 &&

make &&

gcc tools/qt-faststart.c -o tools/qt-faststart
```

HTML documentation was built in the previous step. If you have [texlive-20240312](#) installed and wish to build PDF and Postscript versions of the documentation, issue the following commands:

```

pushd doc &&
for DOCNAME in `basename -s .html *.html` 
do
    texi2pdf -b $DOCNAME.texi &&
    texi2dvi -b $DOCNAME.texi &&

    dvips -o $DOCNAME.ps \
           $DOCNAME.dvi
done &&
popd &&
unset DOCNAME

```

If you have [Doxygen-1.12.0](#) installed and you wish to build (if --disable-doc was used) or rebuild the html documentation, issue:

```
doxygen doc/Doxyfile
```

The fate-suite tests include comparisons with installed files, and should not be run before the package is installed. Therefore, if you desire to run them, instructions are given further below.

Now, as the `root` user:

```

make install &&

install -v -m755 tools/qt-faststart /usr/bin &&
install -v -m755 -d /usr/share/doc/ffmpeg-7.0.2 &&
install -v -m644 doc/*.txt /usr/share/doc/ffmpeg-7.0.2

```

If the PDF and Postscript documentation was built, issue the following commands, as the `root` user, to install them:

```

install -v -m644 doc/*.pdf /usr/share/doc/ffmpeg-7.0.2 &&
install -v -m644 doc/*.ps /usr/share/doc/ffmpeg-7.0.2

```

If you used `doxygen` to manually create the API documentation, install it by issuing the following commands as the `root` user:

```

install -v -m755 -d /usr/share/doc/ffmpeg-7.0.2/api &&
cp -vr doc/doxy/html/* /usr/share/doc/ffmpeg-7.0.2/api &&
find /usr/share/doc/ffmpeg-7.0.2/api -type f -exec chmod -c 0644 \{} \; &&
find /usr/share/doc/ffmpeg-7.0.2/api -type d -exec chmod -c 0755 \{} \;

```

To properly test the installation you must have [rsync-3.3.0](#) installed and follow the instructions for the [FFmpeg Automated Testing Environment](#) (FATE). First, about 1 GB of sample files used to run FATE are downloaded with the command:

```
make fate-rsync SAMPLES=fate-suite/
```

The `fate-suite` directory is created and the files are downloaded there. That command actually runs an rsync command to obtain the sample files. You may want to compress and keep this directory for testing again, for testing in another system, or for when a new version of ffmpeg is released. Then, unpack the sample files in the source directory, and run the `make fate-rsync ...` command above to sync with the upstream repository. The download size and time are drastically reduced by doing this. Estimated values in "Package Information" do not include the download SBU. Some samples may have been removed in newer versions, so in order to be sure local and server fate samples are identical when you use previously saved samples, run the following command:

```

rsync -vrltIw --delete --timeout=60 --contimeout=60 \
rsync://fate-suite.ffmpeg.org/fate-suite/ fate-suite/

```

Next, execute FATE with the following commands (there are more than 3800 tests in the suite):

```

make fate THREADS=N SAMPLES=fate-suite/ | tee ../../fate.log &&
grep ^TEST ../../fate.log | wc -l

```

where `N` is an integer, $N \leq$ number of cores in the system.

A successful run should return no errors or warnings, just a list of tests and total amount at the end.

Command Explanations

`find ... ;`: Fixes permissions of documentation files and directories.

`--enable-libfreetype`: Enables Freetype support.

--enable-gpl: Enables the use of GPL code and permits support for postprocessing, swscale and many other features.

--enable-version3: Enables the use of (L)GPL version 3 code.

--enable-nonfree: Enables the use of nonfree code. Note that the resulting libraries and binaries will be unredistributable.

--disable-static: This switch prevents installation of static versions of the libraries.

--enable-shared: Enables building shared libraries, otherwise only static libraries are built and installed.

--disable-debug: Disables building debugging symbols into the programs and libraries.

--enable-libaom: Enables AV1 audio and video decoding via libaom.

--enable-libass: Enables ASS/SSA subtitle format rendering via libass.

--enable-libdrm: Use this switch if [libdrm-2.4.122](#) is installed to build the "kmsgrab" input module which is useful for screen capturing or streaming.

--enable-libfdk-aac: Enables AAC audio encoding via libfdk-aac.

--enable-libmp3lame: Enables MP3 audio encoding via libmp3lame.

--enable-libvorbis --enable-libvpx: Enables WebM encoding via libvorbis and libvpx.

--enable-libx264: Enables high-quality H.264/MPEG-4 AVC encoding via libx264.

--enable-libx265: Enables high-quality H.265/HEVC encoding via libx265.

--enable-openssl: Enables HTTPS protocol for network streams.

--ignore-tests=...: Disables tests that fail to run on BLFS without the optional packages. Currently the AV1 test is disabled as it needs dav1d to do AV1 decoding on the CPU. The test might work if you have a GPU that can decode AV1, but this was not tested.

--enable-gnutls: Use this option instead of --enable-openssl, if you want to use GnuTLS instead of OpenSSL for HTTPS protocol.

--disable-doc: Disables building html documentation. This is only needed if [Doxygen-1.12.0](#) is installed and you do not want to build the html documentation.

--enable-libpulse: Enables support for Pulseaudio for audio output.

`gcc tools/qt-faststart.c -o tools/qt-faststart`: This builds the `qt-faststart` program which can modify QuickTime formatted movies (.mov or .mp4) so that the header information is located at the beginning of the file instead of the end. This allows the movie file to begin playing before the entire file has been downloaded.

Note

Support for most of the dependency packages requires using options passed to the `configure` script. View the output from `./configure --help` for complete information about enabling dependency packages.

Contents

Installed Programs: ffmpeg, ffplay, fprobe, and qt-faststart

Installed Libraries: libavcodec.so, libavdevice.so, libavfilter.so, libavformat.so, libavutil.so, libpostproc.so, libswresample.so, and libswscale.so

Installed Directories: /usr/include/libav{codec,device,filter,format,util}, /usr/include/libpostproc, /usr/include/libsw{resample,scale}, /usr/share/doc/ffmpeg-7.0.2, and /usr/share/ffmpeg

Short Descriptions

<code>ffmpeg</code>	is a command-line tool to convert video files, network streams and input from a TV card to several video formats
<code>ffplay</code>	is a very simple and portable media player using the <code>ffmpeg</code> libraries and the SDL library
<code>fprobe</code>	gathers information from multimedia streams and prints it in a human and machine-readable fashion
<code>qt-faststart</code>	moves the index file to the front of quicktime (mov/mp4) videos
<code>libavcodec.so</code>	is a library containing the FFmpeg codecs (both encoding and decoding)

libavdevice.so	is the FFmpeg device handling library
libavfilter.so	is a library of filters that can alter video or audio between the decoder and the encoder (or output)
libavformat.so	is a library containing the file formats handling (mux and demux code for several formats) used by <code>ffplay</code> as well as allowing the generation of audio or video streams
libavresample.so	is a library containing functions for resampling audio and video.
libavutil.so	is the FFmpeg utility library
libpostproc.so	is the FFmpeg post processing library
libsresample.so	is the FFmpeg audio rescaling library, it contains functions for converting audio sample formats
libswscale.so	is the FFmpeg image rescaling library

mpv-0.38.0

Introduction to mpv

mpv is a free media player for the command line. It supports a wide variety of media file formats, audio and video codecs, and subtitle types.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mpv-player/mpv/archive/v0.38.0/mpv-0.38.0.tar.gz>
- Download MD5 sum: 7c1a400867e8930f037efc86c1f90a14
- Download size: 3.3 MB
- Estimated disk space required: 28 MB
- Estimated build time: 0.1 SBU (Using parallelism=4)

mpv Dependencies

Required

[alsa-lib-1.2.12](#), [FFmpeg-7.0.2](#), [libass-0.17.3](#), [libplacebo-7.349.0](#), [Mesa-24.1.5](#), and [PulseAudio-17.0](#)

Recommended

[libjpeg-turbo-3.0.1](#), [libva-2.22.0](#), [libvdpau-1.5](#), [Lua-5.2.4](#), [uchardet-0.0.8](#), and [Vulkan-Loader-1.3.294](#)

Optional Input Drivers and Libraries

[libdvdcss-1.4.3](#), [libdvdread-6.1.3](#), [libdvdnav-6.1.1](#), and [libbluray](#)

Optional Audio Output Drivers and Libraries

[pipewire-1.2.3](#), [SDL2-2.30.6](#), [JACK](#), and [OpenAL](#)

Optional Video Output Drivers and Libraries

[libcaca](#) and [SVGAlib](#)

Optional (for documentation)

[docutils-0.21.2](#)

Installation of mpv

Add a fix to build with FFmpeg > 7.0

```
sed -i 's/AV_OPT_TYPE_CHANNEL_LAYOUT/AV_OPT_TYPE_CHLAYOUT/' filters/f_lavfi.c
```

Install mpv by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr      \
            --buildtype=release \
            -D x11=enabled     \
            ..
            &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+ 3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

`-D x11=enabled`: allows linking to the X11 libraries.

`-D dvdnav=enabled`: enables support for DVD playback.

`-D cdda=enabled`: enables support for CD Audio.

`-D sdl2=enabled`: enables the usage of SDL2 for audio and video output.

Configuring mpv

Config Files

`/etc/mpv/*` and `~/.config/mpv/*`

Contents

Installed Programs: mpv

Installed Libraries: None

Installed Directories: /etc/mpv and /usr/share/doc/mpv

Short Descriptions

`mpv` is the mpv video player

VLC-3.0.21

Introduction to VLC

VLC is a media player, streamer, and encoder. It can play from many inputs, such as files, network streams, capture devices, desktops, or DVD, SVCD, VCD, and audio CD. It can use most audio and video codecs (MPEG 1/2/4, H264, VC-1, DivX, WMV, Vorbis, AC3, AAC, etc.), and it can also convert to different formats and/or send streams through the network.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.videolan.org/vlc/3.0.21/vlc-3.0.21.tar.xz>
- Download MD5 sum: cde72f38943c685a1a39acc82da2339f
- Download size: 24 MB
- Estimated disk space required: 746 MB
- Estimated build time: 1.9 SBU (using parallelism=4; with tests)

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/vlc-3.0.21-taglib-1.patch>
- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/vlc-3.0.21-fedora_ffmpeg7-1.patch

VLC Dependencies

Recommended

[alsa-lib-1.2.12](#), [desktop-file-utils-0.27](#), [FFmpeg-7.0.2](#), [iba52-0.8.0](#), [libgcrypt-1.11.0](#) [libmad-0.15.1b](#), [Lua-5.2.4](#), and [a graphical environment](#)

Optional features and packages

[dbus-1.14.10](#), [libidn-1.42](#), and [libssh2-1.11.0](#)

Optional input plugins

[libarchive-3.7.4](#), [libcddb-1.3.2](#), [libdv-1.0.0](#), [libdvdcss-1.4.3](#), [libdvdread-6.1.3](#), [libdvdnav-6.1.1](#), [opencv-4.10.0](#), [Samba-4.20.4](#), [v4l-utils-1.28.1](#), [libbluray](#), [libdc1394](#), [libnfs](#), [libproxy](#), [libraw1394](#), [Live555](#), and [VCDImager](#) (requires [libcdio-2.1.0](#))

Optional mux/demux plugins

[libogg-1.3.5](#), [Game Music Emu](#), [libdvbpsi](#), [libshout](#), [libmatroska](#) (requires [libebml](#)), [libmodplug](#), [Musepack](#), and [sidplay-libs](#)

Optional codec plugins

[FAAD2-2.11.1](#), [FLAC-1.4.3](#), [libao-3.9.1](#), [libass-0.17.3](#), [libmpeg2-0.5.1](#), [libpng-1.6.43](#), [libva-2.22.0](#), [libvorbis-1.3.7](#), [Opus-1.5.2](#), [Speex-1.2.1](#), [libvpx-1.14.1](#), [x264-20240812](#), [aribb24](#), [dav1d](#), [Dirac](#), [FluidLite](#), [FluidSynth](#), [libdca](#), [libkate](#), [libtheora](#), [OpenMAX](#), [Schroedinger](#), [Shine](#), [SoX](#), [Tremor](#), [Twolame](#), and [Zapping VBI](#)

Optional video plugins

[AAlib-1.4rc5](#), [Fontconfig-2.15.0](#), [FreeType-2.13.3](#), [FriBidi-1.0.15](#), [libplacebo-7.349.0](#) (currently broken), [librsvg-2.58.3](#), [libvdpau-1.5](#), [sdl2-compat-1.2.68](#), [libcaca](#), and [libmfx](#)

Optional audio plugins

[PulseAudio-17.0](#), [libsamplerate-0.2.2](#), [spatialaudio](#), and [JACK](#)

Optional interface plugins

[FreeRDP-3.7.0](#), [qt5-components-5.15.14](#) (required for the graphical user interface), [libtar](#), [libvncclient](#), and [LIRC](#)

Optional visualisations and video filter plugins

[Goom](#), [libvsxu](#), and [projectM](#)

Optional service discovery plugins

[Avahi-0.8](#), [libmtp](#) and [libupnp](#)

Miscellaneous options

[GnuTLS-3.8.7.1](#), [libnotify-0.8.3](#), [libxml2-2.13.3](#), [Protobuf-27.3](#), [taglib-2.0.1](#), [xdg-utils-1.2.1](#) (runtime), [AtmoLight](#), [libmicrodns](#), and [Srt](#)

Installation of VLC

First, add corrections for problems of taglib-2.0 and ffmpeg-7 and later:

```
patch -Np1 -i ../../vlc-3.0.21-taglib-1.patch      &&
patch -Np1 -i ../../vlc-3.0.21-fedora_ffmpeg7-1.patch
```

If you wish to add support for lua extensions, enforce the use of lua52:

```
export LUAC=/usr/bin/luac5.2          &&
export LUA_LIBS=$(pkg-config --libs lua52)  &&
export CPPFLAGS=$(pkg-config --cflags lua52)
```

Install VLC by running the following commands:

```
BUILDDCC=gcc ./configure --prefix=/usr --disable-libplacebo &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make docdir=/usr/share/doc/vlc-3.0.21 install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+-3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

`--disable-libplacebo`: This switch is required to disable support for using libplacebo as a rendering pipeline because VLC is incompatible with the latest version of libplacebo.

`--disable-vpx`: Use this switch to disable libvpx.

`--disable-mad`: Use this switch if you don't have libmad installed.

`--disable-avcodec --disable-swscale`: Use these switches if you don't have FFmpeg installed.

`--disable-a52`: Use this switch if you don't have liba52 installed.

`--disable-xcb`: Use this switch if you don't have X Window System installed.

`--disable-alsa`: Use this switch if you don't have ALSA installed.

`--disable-libgcrypt`: Use this switch if you don't have libgcrypt installed.

`--disable-sftp`: The default is to build the sftp module if [libssh2-1.11.0](#) is installed. Use this switch if you do not want to build this module.

Note

There are many options to VLC's `configure` command. Check the `configure --help` output for a complete list.

Contents

Installed Programs: cvlc, nvlc, qvlc, rvlc, svlc, vlc and vlc-wrapper

Installed Libraries: libvlccore.so, libvlc.so, and many plugins in /usr/lib/vlc/plugins

Installed Directories: /usr/{include,lib,share}/vlc, /usr/share/kde4, and /usr/share/doc/vlc-3.0.21

Short Descriptions

<code>cvlc</code>	is a script to run VLC with the dummy interface
<code>nvlc</code>	is a script to run VLC with the ncurses interface
<code>qvlc</code>	is a script to run VLC with the Qt interface
<code>rvlc</code>	is a script to run VLC with a command line interface
<code>svlc</code>	is a script to run VLC with the skins interface
<code>vlc</code>	is the VLC media player
<code>vlc-wrapper</code>	is a wrapper to drop privileges with VLC

xine-ui-0.99.14

Introduction to Xine User Interface

The xine User Interface package contains a multimedia player. It plays back CDs, DVDs and VCDs. It also decodes multimedia files like AVI, MOV, WMV, MPEG and MP3 from local disk drives, and displays multimedia streamed over the Internet.

This package is known to build and work properly using an LFS 12.2 platform.

Note

Although this version of xine-ui works with most files, it is unable to open encrypted (content-scrambled) DVDs with the current version of libdvdcss.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/xine/xine-ui-0.99.14.tar.xz>
- Download MD5 sum: 86a4db9050405a91fcc33b7ad85274f5
- Download size: 2.1 MB
- Estimated disk space required: 23 MB
- Estimated build time: 0.2 SBU

Xine User Interface Dependencies

Required

[xine-lib-1.2.13](#) and [shared-mime-info-2.4](#)

Optional

[cURL-8.9.1](#), [AAlib-1.4rc5](#), [LIRC](#), and [libcaca](#)

Installation of Xine User Interface

Install xine User Interface by running the following commands:

```
./configure --prefix=/usr &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make docsdir=/usr/share/doc/xine-ui-0.99.14 install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating

/usr/share/icons/hicolor/index.theme and /usr/share/applications/mimeinfo.cache. To perform the update you must have [GTK+ 3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

`docsdir=/usr/share/doc/xine-ui-0.99.14`: This parameter causes the Xine UI documentation to be installed in the versioned directory `/usr/share/doc/xine-ui-0.99.14`, rather than the default `/usr/share/doc/xine-ui`.

Configuring Xine User Interface

Config Files

`~/.xine/config`

Configuration Information

The above file is created and maintainable through the `xine` setup dialog box. The documentation for the configuration settings is located at `/usr/share/doc/xine-ui-0.99.14/README.config_en`.

If you have a DVB TV card, you can watch TV with the command `xine dvb://` and change channels with the scroll wheel on your mouse.

Contents

Installed Programs: aaxine, cacaxine, fbxine, xine, xine-bugreport, xine-check, and xine-remote

Installed Libraries: None

Installed Directories: /usr/share/xine and /usr/share/doc/xine-ui-0.99.14

Short Descriptions

<code>aaxine</code>	is an ASCII art video player which utilizes AAlib as the frontend for the xine Libraries
<code>cacaxine</code>	is a color ASCII art video player which utilizes CACA as the frontend for the xine Libraries
<code>fbxine</code>	is a frame buffer interface to the xine Libraries
<code>xine</code>	is a multimedia player designed to play MPEG streams (audio and video), MPEG elementary streams (MP3), MPEG transport streams, Ogg files, AVI files, ASF files, some Quicktime files, VCDs and DVDs
<code>xine-bugreport</code>	produces a terse system description and guides you through the process of reporting a bug
<code>xine-check</code>	tests the <code>xine</code> video player installation for common problems. It tests the operating system settings, installation of plugins, CD/DVD drive settings and video support parameters
<code>xine-remote</code>	is a tool to connect to a <code>xine</code> remote control server

Chapter 45. CD/DVD-Writing Utilities

This chapter contains information on CD/DVD-writing utilities in Linux.

Additional sources of information include:

- [CD-Writing HOWTO](#)
- [CD-Recordable FAQ](#)
- [The dvd+rw-tools Website](#)

Cdrdao-1.2.4

Introduction to Cdrdao

The Cdrdao package contains CD recording utilities. These are useful for burning a CD in disk-at-once mode.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/cdrdao/cdrdao-1.2.4.tar.bz2>
- Download MD5 sum: 2ada887d1b30b440867b8df0d3023cf7
- Download size: 1.4 MB
- Estimated disk space required: 39 MB
- Estimated build time: 0.1 SBU

Cdrdao Dependencies

Recommended

[libao-1.2.0](#), [libvorbis-1.3.7](#), [libmad-0.15.1b](#), and [LAME-3.100](#) (required to build toc2mp3)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/cdrdao>

Installation of Cdrdao

Install Cdrdao by running the following commands:

```
./configure --prefix=/usr --mandir=/usr/share/man &&  
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install &&  
install -v -m755 -d /usr/share/doc/cdrdao-1.2.4 &&  
install -v -m644 README /usr/share/doc/cdrdao-1.2.4
```

Command Explanations

`--mandir=/usr/share/man`: Install manual pages in `/usr/share/man` instead of `/usr/man`.

Contents

Installed Programs: cdrdao, cue2toc, toc2cddb, toc2cue and optionally toc2mp3

Installed Libraries: None

Installed Directories: /usr/share/cdrdao, /usr/share/doc/cdrdao-1.2.4

Short Descriptions

<code>cdrdao</code>	records audio or data CD-Rs in disk-at-once (DAO) mode based on a textual description of the CD contents
<code>cue2toc</code>	converts CUE to TOC format for audio CDs
<code>toc2cddb</code>	converts a Cdrdao TOC file into a cddb file and prints it to stdout
<code>toc2cue</code>	converts TOC to CUE format for audio CDs
<code>toc2mp3</code>	converts an audio CD disk image (<code>.toc</code> file) to MP3 files

Cdrtools-3.02a09

Introduction to Cdrtools

The Cdrtools package contains CD recording utilities. These are useful for reading, creating or writing (burning) CDs, DVDs, and Blu-ray discs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/cdrtools/cdrtools-3.02a09.tar.bz2>

- Download MD5 sum: 1e224a6adbbe4ad40047b9fddb0e60c
- Download size: 2.1 MB
- Estimated disk space required: 32 MB
- Estimated build time: 0.7 SBU

Cdrtools Dependencies

Required

[alsa-lib-1.2.12](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/Cdrtools>

Installation of Cdrtools

Note

This package does not support parallel build.

Install Cdrtools by running the following commands:

```
sed -i 's!-/opt/schily!-/usr!g'          DEFAULTS/Defaults.linux &&
sed -i 's!DEFINSGRP=.*!DEFINSGRP=root!'  DEFAULTS/Defaults.linux &&
sed -i 's!INSDIR=\$*sbin!INSDIR=bin!'    rscsi/Makefile      &&

export GMAKE_NOWARN=true &&
export CFLAGS="$CFLAGS -std=gnu89 -fno-strict-aliasing" &&

make -j1 INS_BASE=/usr \
      DEFINUSR=root \
      DEFINSGRP=root \
      VERSION_OS="LinuxFromScratch"
```

This package does not come with a test suite.

Now, as the `root` user:

```
GMAKE_NOWARN=true &&

make INS_BASE=/usr \
      DEFINUSR=root \
      DEFINSGRP=root \
      MANSUFF_LIB=3cdr \
      install      &&

install -v -m755 -d /usr/share/doc/cdrtools-3.02a09 &&
install -v -m644 README.* READMEs/* ABOUT doc/*.ps \
          /usr/share/doc/cdrtools-3.02a09
```

Command Explanations

`export GMAKE_NOWARN=true`: This variable avoids a warning when using GNU make.

`INS_BASE=/usr`: This parameter moves the install directory from `/opt/schily` to `/usr`.

`DEFINUSR=root DEFINSGRP=root`: These parameters install all programs with root:root ownership instead of the default bin:bin.

`MANSUFF_LIB=3cdr`: This parameter prevents Cdrtools from overwriting important pages which are already installed by LFS. The installed man pages of section 3 will have a suffix `3cdr` and can be shown using `man 3cdr <manpage>`.

Contents

Installed Programs: `btcflash`, `cdda2mp3`, `cdda2ogg`, `cdda2wav`, `cdrecord`, `devdump`, `isodebug`, `isodump`, `isoinfo`, `isovfy`, `mkyhybrid`, `mkisofs`, `readcd`, `rscsi`, `scgcheck`, `scgskeleton`

Installed Libraries: `libcdrdeflt.a`, `libdeflt.a`, `libedc_ecc.a`, `libedc_ecc_dec.a`, `libfile.a`, `libfind.a`, `libhfs.a`, `libmdigest.a`, `libparanoia.a`, `librscg.a`, `libscg.a`, `libscgcmd.a`, `libschily.a`, `libsconv.a`

Installed Directories: /usr/lib/{profiled,sconv}, /usr/include/{scg,schily},
/usr/share/doc/cdda2wav,cdrecord,libparanoia,mkisofs,rscsi}

Short Descriptions

btcflash	flashes the firmware on BTC DRW1008 DVD+/-RW recorders. Please exercise care with this program
cdda2wav	converts Compact Disc audio into WAV sound files
cdrecord	records audio or data onto Compact Discs
devdump	is a diagnostic program used to dump an ISO-9660 device or file in hex
isodebug	is used to display the command-line parameters used to create an ISO-9660 image
isodump	is a diagnostic program used to dump a device or file based on ISO-9660
isoinfo	is used to analyze or list an ISO-9660 image
isovfy	is used to verify an ISO-9660 image
mkhybrid	is a symbolic link to <code>mkisofs</code> used to create ISO-9660/HFS hybrid filesystem images
mkisofs	is used to create ISO-9660/JOLIET/HFS filesystem images, optionally with Rock Ridge attributes
readcd	reads or writes Compact Discs
rscsi	is a remote SCSI manager
scgcheck	is used to check and verify the Application Binary Interface of <code>libscg</code>
libscg.a	is a highly portable SCSI transport library

dvd+rw-tools-7.1

Introduction to dvd+rw-tools

The `dvd+rw-tools` package contains several utilities to master the DVD media, both +RW/+R and -R[W]. The principle tool is `growisofs` which provides a way to both lay down **and** grow an ISO9660 file system on (as well as to burn an arbitrary pre-mastered image to) all supported DVD media. This is useful for creating a new DVD or adding to an existing image on a partially burned DVD.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <http://fy.chalmers.se/~appro/linux/DVD+RW/tools/dvd+rw-tools-7.1.tar.gz>
- Download MD5 sum: 8acb3c885c87f6838704a0025e435871
- Download size: 138 KB
- Estimated disk space required: 1.7 MB
- Estimated build time: less than 0.1 SBU

dvd+rw-tools Dependencies

Required

[Cdrtools-3.02a09](#) (runtime)

Installation of dvd+rw-tools

Install `dvd+rw-tools` by running the following commands:

```
sed -i '/stat.h/a #include <sys/sysmacros.h>' growisofs.c &&
sed -i '/stdlib/a #include <limits.h>' transport.hxx &&
make all rpl8 btcflash
```

This package does not come with a test suite.

Now, as the `root` user:

```
make prefix=/usr install &&
install -v -m644 -D index.html \
/usr/share/doc/dvd+rw-tools-7.1/index.html
```

Command Explanations

```
sed -i '/stdlib/a #include <limits.h>' ...: This sed includes limits.h, one of the kernel headers. This is needed due to a change in the 2.6.23 kernel headers.
```

```
make all rp18 btcflash: This command uses additional targets so that all the utilities are built.
```

Contents

Installed Programs: btcflash, dvd+rw-booktype, dvd+rw-format, dvd+rw-mediainfo, dvd-ram-control, growisofs, and rp18

Installed Libraries: None

Installed Directory: /usr/share/doc/dvd+rw-tools-7.1

Short Descriptions

`growisofs` is a combined `mkisofs` frontend/DVD recording program

libburn-1.5.6

Introduction to libburn

libburn is a library for writing preformatted data onto optical media: CD, DVD and BD (Blu-Ray).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.libburnia-project.org/releases/libburn-1.5.6.tar.gz>
- Download MD5 sum: cf9852f3b71dbc2b6c9e76f6eb0474f0
- Download size: 972 KB
- Estimated disk space required: 17 MB
- Estimated build time: 0.1 SBU

libburn Dependencies

Optional

[Doxygen-1.12.0](#) (to generate HTML documentation)

Installation of libburn

Install libburn by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

If you have installed Doxygen and wish to generate the HTML documentation, issue the following command:

```
doxygen doc/doxygen.conf
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have built the HTML documentation, install it by running the following commands as the `root` user:

```
install -v -dm755 /usr/share/doc/libburn-1.5.6 &&
install -v -m644 doc/html/* /usr/share/doc/libburn-1.5.6
```

Contents

Installed Program: cdrskin

Installed Library: libburn.so

Installed Directory: /usr/include/libburn

Short Descriptions

cdrskin	burns preformatted data to CD, DVD, and BD via libburn
libburn.so	contains the libburn API functions

libisoburn-1.5.6

Introduction to libisoburn

libisoburn is a frontend for the libburn and libisofs libraries, which enables creation and expansion of ISO-9660 filesystems on all CD/DVD/BD media supported by libburn. This includes media like DVD+RW, which do not support multi-session management on the media level, and even plain disk files or block devices.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.libburnia-project.org/releases/libisoburn-1.5.6.tar.gz>
- Download MD5 sum: efb19f7f718f0791f717b2c6094995ec
- Download size: 1.5 MB
- Estimated disk space required: 20 MB (with Tk and HTML documentation)
- Estimated build time: 0.2 SBU (with Tk and HTML documentation)

libisoburn Dependencies

Required

[libburn-1.5.6](#) and [libisofs-1.5.6](#)

Optional

[Doxygen-1.12.0](#) (to generate HTML documentation), [Tk-8.6.14](#) (for `xorriso-tcltk`), and [libjte](#)

Installation of libisoburn

Install libisoburn by running the following commands:

```
./configure --prefix=/usr          \
            --disable-static      \
            --enable-pkg-check-modules &&
make
```

If you have installed Doxygen and wish to generate the HTML documentation, issue the following command:

```
doxygen doc/doxygen.conf
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have built the HTML documentation, install it by running the following commands as the `root` user:

```
install -v -dm755 /usr/share/doc/libisoburn-1.5.6 &&
install -v -m644 doc/html/* /usr/share/doc/libisoburn-1.5.6
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-pkg-check-modules`: This switch tells the configure script to use pkg-config to check for the libburn and libisofs libraries.

Contents

Installed Programs: osirrox, xorrecord, xorriso, xorriso-dd-target, xorrisofs, and xorriso-tcltk

Installed Library: libisoburn.so

Installed Directories: /usr/include/libisoburn and /usr/share/doc/libisoburn-1.5.6

Short Descriptions

<code>osirrox</code>	is a symbolic link to <code>xorriso</code> that copies files from ISO image to a disk filesystem
<code>xorrecord</code>	is a symbolic link to <code>xorriso</code> that provides a cdrecord type user interface
<code>xorriso</code>	is a program to create, load, manipulate, read, and write ISO 9660 filesystem images with Rock Ridge extensions
<code>xorriso-dd-target</code>	is a program to check a USB or memory card device to see whether it is suitable for image copying
<code>xorrisofs</code>	is a symbolic link to <code>xorriso</code> that provides a mkisofs type user interface
<code>xorriso-tcltk</code>	is a frontend that operates xorriso in dialog mode
<code>libisoburn.so</code>	contains the libisoburn API functions

libisofs-1.5.6

Introduction to libisofs

libisofs is a library to create an ISO-9660 filesystem with extensions like RockRidge or Joliet.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.libburnia-project.org/releases/libisofs-1.5.6.tar.gz>
- Download MD5 sum: 9f996b317f622802f12d28d27891709f
- Download size: 858 KB
- Estimated disk space required: 14 MB
- Estimated build time: 0.1 SBU

libisofs Dependencies

Optional

[Doxygen-1.12.0](#) (to generate HTML documentation) and [libjte](#)

Installation of libisofs

Install libisofs by running the following commands:

```
./configure --prefix=/usr --disable-static &&
make
```

If you have installed Doxygen and wish to generate the HTML documentation, issue the following command:

```
doxygen doc/doxygen.conf
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

If you have built the HTML documentation, install it by running the following commands as the `root` user:

```
install -v -dm755 /usr/share/doc/libisofs-1.5.6 &&
install -v -m644 doc/html/* /usr/share/doc/libisofs-1.5.6
```

Command Explanations

--disable-static: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: None

Installed Library: libisofs.so

Installed Directory: /usr/include/libisofs

Short Descriptions

libisofs.so contains the libisofs API functions

Part XIII. Printing, Scanning and Typesetting

Chapter 46. Printing

This chapter contains spooling printer management systems and ghostscript applications to render PostScript for display on terminals or paper.

Cups-2.4.10

Introduction to Cups

The Common Unix Printing System (CUPS) is a print spooler and associated utilities. It is based on the "Internet Printing Protocol" and provides printing services to most PostScript and raster printers.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/OpenPrinting/cups/releases/download/v2.4.10/cups-2.4.10-source.tar.gz>
- Download MD5 sum: 0e8b1227340e4aec22b9fbbcd6ca5cd
- Download size: 7.8 MB
- Estimated disk space required: 60 MB (Add 49 MB for tests)
- Estimated build time: 0.4 SBU (Add 1.0 SBU for tests)

Cups Dependencies

Required

[GnuTLS-3.8.7.1](#)

Recommended

[colord-1.4.7](#), [dbus-1.14.10](#), [libusb-1.0.27](#), [Linux-PAM-1.6.1](#), and [xdg-utils-1.2.1](#)

Optional

[Avahi-0.8](#), [libpaper-2.2.5](#), [MIT Kerberos V5-1.21.3](#), [PHP-8.3.10](#), and [Python2](#)

Required (Postinstall)

[cups-filters-2.0.1](#)

Optional (Postinstall)

[Gutenprint-5.3.4](#) and [hplip](#) (HP printers)

Kernel Configuration

Note

There used to be a conflict between the Cups libusb backend and the usblp kernel driver. This is no longer the case and cups will work with both of these enabled.

If you want to use the kernel usblp driver (for example, if you wish to use escputil from [Gutenprint-5.3.4](#)) enable the following options in your kernel configuration and recompile the kernel:

```
Device Drivers --->
  [*] USB support --->
    <*/M> USB Printer support                               [USB_SUPPORT]
                                                    [USB_PRINTER]
```

If you have a parallel printer, enable the following options in your kernel configuration and recompile the kernel:

```
Device Drivers --->
  <*/M> Parallel port support --->
    <*/M> PC-style hardware                               [PARPORT]
  Character devices --->
    <*/M> Parallel printer support                         [PARPORT_PC]
                                                    [PRINTER]
```

Installation of Cups

You will need to add an `lp` user, as Cups will create some files owned by this user. (The `lp` user is the default used by Cups, but may be changed to a different user by passing a parameter to the `configure` script.) Use the following command as the `root` user:

```
useradd -c "Print Service User" -d /var/spool/cups -g lp -s /bin/false -u 9 lp
```

You will also need a dedicated group that will contain users allowed to do Cups administrative tasks. Add the group by running the following command as the `root` user:

```
groupadd -g 19 lpadmin
```

If you want to add a user to the Cups administrative group, run the following command as the `root` user:

```
usermod -a -G lpadmin <username>
```

If you didn't install [xdg-utils-1.2.1](#), use the following `sed` to change the default browser that will be used to access the Cups web interface:

```
sed -i 's#@CUPS_HTMLVIEW@#firefox#' desktop/cups.desktop.in
```

Replace `firefox` with the web browser of your choice.

Build Cups by running the following commands:

```
./configure --libdir=/usr/lib \
--with-rundir=/run/cups \
--with-system-groups=lpadmin \
--with-docdir=/usr/share/cups/doc-2.4.10 &&
make
```

To test the results, issue: `LC_ALL=C make -k check`. An already active graphical session with bus address is necessary to run the tests. Make sure that there is not other instance of Cups running, otherwise at least 4 tests will fail with "address in use".

Now, as the `root` user:

```
make install &&
ln -svnf ../cups/doc-2.4.10 /usr/share/doc/cups-2.4.10
```

Create a basic Cups client configuration file by running the following command as the `root` user:

```
echo "ServerName /run/cups/cups.sock" > /etc/cups/client.conf
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and you can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme`. To perform the update you must have [GTK+-3.24.43](#) installed and issue the following command as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor
```

Command Explanations

- with-rundir=/run/cups: This switch prevents the package from referring to the deprecated `/var/run` directory.
- with-system-groups=lpadmin: This switch ensures that only `lpadmin` will be used as the Cups administrative group.
- disable-libusb: Use this switch if you have installed [libusb-1.0.27](#), but wish to use the kernel usblp driver.
- enable-libpaper: Use this switch if you have installed libpaper and wish to use it with Cups.

Configuring Cups

Config Files

```
/etc/cups/*
```

Configuration Information

Normally, printers are set up via a web browser. The Cups server will normally connect via the url `http://localhost:631`. From there printers, print jobs, and the server configuration can be set up and managed. Remote system administration can also be set up. Configuration can also be done from the command line via the `lpadmin`, `lpoptions`, and `lpstat` commands.

Configuration of Cups is dependent on the type of printer and can be complex. Generally, PostScript printers are easier. For detailed instructions on configuration and use of Cups, see <https://www.cups.org/documentation.html>.

For non-PostScript printers to print with Cups, you need to install [ghostscript-10.03.1](#) to convert PostScript to raster images and a driver (e.g. from [Gutenprint-5.3.4](#)) to convert the resulting raster images to a form that the printer understands. [Foomatic](#) drivers use [ghostscript-10.03.1](#) to convert PostScript to a printable form directly, but this is considered suboptimal by Cups developers.

Linux PAM Configuration

If CUPS has been built with Linux PAM support, you need to create a PAM configuration file to get it working correctly with BLFS.

Issue the following command as the `root` user to create the configuration file for Linux PAM:

```
cat > /etc/pam.d/cups << "EOF"
# Begin /etc/pam.d/cups

auth    include system-auth
account include system-account
session include system-session

# End /etc/pam.d/cups
EOF
```

Systemd Unit

To start the `cupsd` daemon when something tries to access it, enable the previously installed systemd units by running the following command as the `root` user:

```
systemctl enable cups
```

Contents

- Installed Programs:** cancel, cupsaccept, cups-config, cupsctl, cupsd, cupsdisable, cupsenable, cupsfilter, cupsreject, cupstestppd, ippeveprinter, ippfind, ippool, lp, lpadmin, lpc, lpinfo, lpmove, lpoptions, lpq, lpr, lprm, lpstat, ppdc, ppdhtml, ppdi, ppdmerge, and ppdo
- Installed Libraries:** libcupsimage.so and libcups.so
- Installed Directories:** /etc/cups, /usr/{include,lib,share}/cups, /usr/share/doc/cups-2.4.10, /run/cups, and /var/{cache,log,spool}/cups

Short Descriptions

<code>cancel</code>	cancels existing print jobs from the print queues
<code>cupsaccept</code>	accepts jobs sent to a destination
<code>cups-config</code>	is a CUPS program configuration utility
<code>cupstctl</code>	updates or queries the cupsd.conf file for a server
<code>cupsd</code>	is the scheduler for the Common Unix Printing System
<code>cupsdisable</code>	stops printers and classes
<code>cupsenable</code>	starts printers and classes
<code>cupsfilter</code>	is a front-end to the CUPS filter subsystem which allows you to convert a file to a specific format
<code>cupsreject</code>	rejects jobs sent to a destination
<code>cupstestppd</code>	tests the conformance of PPD files
<code>ippforallprinters</code>	is an IPP Everywhere printer for CUPS
<code>ippfind</code>	finds Internet Printing Protocol printers
<code>ipptool</code>	sends IPP requests to the specified URI and tests and/or displays the results
<code>lp</code>	submits files for printing or alters a pending job
<code>lpadmin</code>	configures printer and class queues provided by CUPS
<code>lpc</code>	provides limited control over printer and class queues provided by CUPS
<code>lpinfo</code>	lists the available devices or drivers known to the CUPS server
<code>lpmove</code>	moves the specified job to a new destination
<code>lpoptions</code>	displays or sets printer options and defaults
<code>lpq</code>	shows the current print queue status on the named printer
<code>lpr</code>	submits files for printing
<code>lprm</code>	cancels print jobs that have been queued for printing
<code>lpstat</code>	displays status information about the current classes, jobs, and printers
<code>ppdc</code>	compiles PPDC source files into one or more PPD files
<code>ppdhtml</code>	reads a driver information file and produces a HTML summary page that lists all of the drivers in a file and the supported options
<code>ppdi</code>	imports one or more PPD files into a PPD compiler source file
<code>ppdmmerge</code>	merges two or more PPD files into a single, multi-language PPD file
<code>ppdpo</code>	extracts UI strings from PPDC source files and updates either a GNU gettext or Mac OS X strings format message catalog source file for translation
<code>libcups.so</code>	contains the CUPS API functions

cups-browsed-2.0.1

Introduction to cups-browsed

The cups-browsed daemon is used to browse the network for remote CUPS queues and IPP network printers and automatically create local queues pointing to them.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/OpenPrinting/cups-browsed/releases/download/2.0.1/cups-browsed-2.0.1.tar.xz>
- Download MD5 sum: b6b5bad9b3bb52afb687702e0b3f6d02
- Download size: 420 KB
- Estimated disk space required: 6.1 MB
- Estimated build time: less than 0.1 SBU (Using parallelism=4)

cups-browsed Dependencies

Required

[Avahi-0.8](#), [libcupsfilters-2.0.0](#) and [libppd-2.0.0](#)

Optional

[OpenLDAP-2.6.8](#)

Installation of cups-browsed

Install cups-browsed by running the following commands:

```
./configure --prefix=/usr          \
--with-cups-rundir=/run/cups    \
--without-rcdir                 \
--disable-static                 \
--docdir=/usr/share/doc/cups-browsed-2.0.1 &&
make
```

This package does not come with a functional test suite.

Now, as the `root` user:

```
make install
```

Install the systemd unit by running the following command as the `root` user:

```
install -v -m644 daemon/cups-browsed.service /lib/systemd/system/cups-browsed.service
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-cups-rundir=/run/cups`: Defines the directory containing transient printing data. The default `/var/run/cups` is obsolete.

`--without-rcdir`: This switch disables installation of the bundled bootscript which is not compatible with BLFS.

Configuring cups-browsed

Systemd Unit

To start the `cups-browsed` daemon at boot, enable the previously installed systemd unit by running the following command as the `root` user:

```
systemctl enable cups-browsed
```

Contents

Installed Programs: `cups-browsed`

Installed Libraries: None

Installed Directories: None

Short Descriptions

`cups-browsed` is a daemon for browsing the Bonjour and CUPS broadcasts of shared, remote CUPS printers

cups-filters-2.0.1

Introduction to CUPS Filters

The CUPS Filters package is only needed to adapt [libcupsfilters-2.0.0](#) and [libppd-2.0.0](#) filter functions for cups-2.x. It should not be needed for cups-3.x.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/OpenPrinting/cups-filters/releases/download/2.0.1/cups-filters-2.0.1.tar.xz>
- Download MD5 sum: b40d0f135af414666e1c3917983a1bb3
- Download size: 440 KB

- Estimated disk space required: 9.0 MB
- Estimated build time: less than 0.1 SBU (Using parallelism=4)

CUPS Filters Dependencies

Required

[libcupsfilters-2.0.0](#) and [libppd-2.0.0](#)

Optional (runtime)

[cups-browsed-2.0.1](#) (if you need Network Discovery capabilities)

Installation of CUPS Filters

Install CUPS Filters by running the following commands:

```
./configure --prefix=/usr          \
            --disable-static      \
            --docdir=/usr/share/doc/cups-filters-2.0.1 &&
make
```

This package does not come with a functional test suite.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

Contents

Installed Programs: driverless, driverless-fax, and foomatic-rip

Installed Libraries: None

Installed Directories: /usr/lib/cups/{backend,driver,filter}, /usr/share/doc/cups-filters-2.0.1, and /usr/share/ppd/cupsfilters

Short Descriptions

<code>driverless</code>	is a PPD generator utility for driverless printing
<code>driverless-fax</code>	is a wrapper to <code>driverless</code> for FAX type devices
<code>foomatic-rip</code>	is a universal print filter/RIP wrapper which can be used as CUPS filter or stand-alone for spooler-less, direct printing

ghostscript-10.03.1

Introduction to Ghostscript

Ghostscript is a versatile processor for PostScript data with the ability to render PostScript to different targets. It is a mandatory part of the cups printing stack.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/ArtifexSoftware/ghostpdl-downloads/releases/download/gs10031/ghostscript-10.03.1.tar.xz>
- Download MD5 sum: 248294abc5aee3ca8161012853d73b52
- Download size: 65 MB
- Estimated disk space required: 291 MB (with fonts and libgs.so)
- Estimated build time: 1.2 SBU (Using parallelism=4; with libgs.so)

Additional Downloads

If you wish, you can download additional fonts.

Standard Fonts

- Download (HTTP): <https://downloads.sourceforge.net/gs-fonts/ghostscript-fonts-std-8.11.tar.gz>
- Download MD5 sum: 6865682b095f8c4500c54b285ff05ef6
- Download size: 3.7 MB

Other Fonts

- Download (HTTP): <https://downloads.sourceforge.net/gs-fonts/gnu-gs-fonts-other-6.0.tar.gz>
- Download MD5 sum: 33457d3f37de7ef03d2eea05a9e6aa4f
- Download size: 796 KB

Ghostscript Dependencies

Recommended

[Cups-2.4.10](#) (required for building the “cups” device backend, which is needed by cups-filters), [Fontconfig-2.15.0](#) (required if you are installing any of the suggested fonts), [FreeType-2.13.3](#), [Little CMS-2.16](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), [libtiff-4.6.0](#), and [OpenJPEG-2.5.2](#)

Optional

[Cairo-1.18.0](#), [GTK+-3.24.43](#), [libidn-1.42](#), [libpaper-2.2.5](#), [libwebp-1.4.0](#), and [a graphical environment](#)

Installation of Ghostscript

Note

The Ghostscript build system is not user-friendly. In order to use system copies of various graphics libraries, you must do it using unconventional methods.

Ghostscript includes (old) copies of several libraries. Some of these seem to have been patched to fix known vulnerabilities, but others of these copies are less-well maintained. To ensure that any future fixes are applied throughout the whole system, it is recommended that you first install the released versions of these libraries and then configure Ghostscript to link to them.

If you have installed the recommended dependencies on your system, remove the copies of freetype, lcms2, libjpeg, libpng, and openjpeg:

```
rm -rf freetype lcms2mt jpeg libpng openjpeg
```

Fix building this package with gcc-14 and libidn:

```
sed -e '186 s/NewPassword /*NewPassword =' \
      -e '187 s/NewLen /*NewLen =' \
      -i pdf/pdf_sec.c
```

Compile Ghostscript:

```
rm -rf zlib &&
./configure --prefix=/usr           \
            --disable-compile-init \
            --with-system-libtiff  &&
make
```

Note

The shared library depends on [GTK+-3.24.43](#). It is only used by external programs like [asymptote-2.91](#), [dvisvgm-3.4](#), and [ImageMagick-7.1.1-36](#).

To compile the shared library `libgs.so`, run the following additional command as an unprivileged user:

```
make so
```

This package does not come with a test suite. A set of example files may be used for testing, but it is only possible after installation of the package.

Now, as the `root` user:

```
make install
```

If you built the shared library, install it with:

```
make soinstall           &&
install -v -m644 base/*.* /usr/include/ghostscript &&
ln -sfv ghostscript /usr/include/ps
```

Now make the documentation accessible from a standard place:

```
mv -v /usr/share/doc/ghostscript/10.03.1 /usr/share/doc/ghostscript-10.03.1 &&
rmdir /usr/share/doc/ghostscript                                &&
cp -r examples/ -T /usr/share/ghostscript/10.03.1/examples
```

If you have downloaded the fonts, unpack them to `/usr/share/ghostscript` and ensure the ownership of the files are `root`:

```
tar -xvf ../ghostscript-fonts-std-8.11.tar.gz -C /usr/share/ghostscript --no-same-owner &&
tar -xvf ../gnu-gs-fonts-other-6.0.tar.gz      -C /usr/share/ghostscript --no-same-owner &&
fc-cache -v /usr/share/ghostscript/fonts/
```

You can now test the rendering of various postscript and pdf files from the `/usr/share/ghostscript/10.03.1/examples`. To do this, run the following command (in a X11 session):

```
gs -q -dBATCH /usr/share/ghostscript/10.03.1/examples/tiger.eps
```

Command Explanations

`rm -rf zlib` : zlib was installed as part of LFS.

`--disable-compile-inits`: This option makes `gs` and `libgs.so` slightly smaller.

`--with-system-libtiff`: Remove this option if you've not installed [libtiff-4.6.0](#).

`install -v -m644 base/*.*...` : Some packages (ImageMagick is one) need the Ghostscript interface headers in place to link to the shared library. These commands install the headers.

`ln -sfv ghostscript /usr/include/ps`: Some packages expect to find the interface headers in an alternate location.

`mv -v /usr/share/doc/ghostscript/10.03.1 /usr/share/doc/ghostscript-10.03.1`: This moves the documentation to where it is expected to be found.

`--disable-cups`: This option will save a tiny amount of space by not linking `gs` and `libgs.so` to the [Cups-2.4.10](#) libraries if you have installed those.

Contents

Installed Programs: dvipdf, eps2eps, gs, gsbj, gsc (from soinstall), gsdj, gsdj500, gslj, gslp, gsnd, gsx (from soinstall), lprsetup.sh, pdf2dsc, pdf2ps, pf2afm, pfbtopfa, pphs, printafm, ps2ascii, ps2epsi, ps2pdf, ps2pdf12, ps2pdf13, ps2pdf14, ps2pdfwr, ps2ps, ps2ps2, and unix-lpr.sh

Installed Library: libgs.so

Installed Directories: /usr/include/ghostscript, /usr/share/ghostscript, and /usr/share/doc/ghostscript-10.03.1

Short Descriptions

`gs` is an interpreter for Adobe Systems' PostScript(tm) and Portable Document Format (PDF)
`libgs.so` provides Ghostscript functionality to other programs, such as GSView , ImageMagick , and libspectre

Ghostscript provides many different scripts used to convert PostScript, PDF, and other formats. Please refer to the HTML documentation or the man pages for information about the capabilities provided.

Gutenprint-5.3.4

Introduction to Gutenprint

The Gutenprint (formerly Gimp-Print) package contains high quality drivers for many brands and models of printers for use with [Cups-2.4.10](#). See a list of supported printers at https://gutenprint.sourceforge.net/p_Supported_Printers.php.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/gimp-print/gutenprint-5.3.4.tar.xz>
- Download MD5 sum: dc208cfb873f13a93439185c15aa6f81
- Download size: 5.1 MB
- Estimated disk space required: 59 MB (without tests)
- Estimated build time: 0.4 SBU (without tests)

Gutenprint Dependencies

Recommended

[Cups-2.4.10](#)

Recommended (at runtime)

Two families of DyeSub photo printers each require another library at runtime. Unless you have one of these devices you will not want these libraries.

For the Sinfonia S6145 and the Ciat Brava 21, [libS6145ImageReProcess](#), see [Peachy Photos \(Sinfonia\)](#) for instructions on how to build and test this.

For the Mitsubishi D70 family and the similar Kodak 305, [libMitsuD70ImageReProcess](#), see [Peachy Photos \(Mitsubishi CP-D70\)](#) for instructions on how to build and test this.

Optional (to Regenerate Documentation)

[ImageMagick-7.1.1-36](#), [texlive-20240312](#) (or [install-tl-unx](#)), [Doxygen-1.12.0](#), and [DocBook-utils-0.6.14](#)

Kernel Configuration

If you intend to use escputil with a USB-connected Epson Stylus printer for commands such as `--ink-level` which need to access the raw device, you must enable the kernel's usblp driver. Enable the following options in your kernel configuration and recompile the kernel:

```
Device Drivers --->
  [*] USB support --->
    <*/M> USB Printer support                               [USB_SUPPORT]
                                                 [USB_PRINTER]
```

Installation of Gutenprint

Install Gutenprint by running the following commands:

```
sed -i 's|${PACKAGE}/doc|doc/${PACKAGE}-${VERSION}|' \
      ,doc/,doc/developer/}Makefile.in &&

./configure --prefix=/usr \
            --disable-static \
            --without-gimp2 \
            --without-gimp2-as-gutenprint &&
make
```

To test the results, issue: `make check`. The tests take a very long time and use a lot of disk space. When last tested (version 5.2.15 in June 2019) this needed 75 SBU and 13.5 GB to complete the tests (one test, `rastertogutenprint` failed because

the CUPS ppds are for a previous version. Maybe genppds should be run before the tests). Note that the tests used only one CPU.

Now, as the `root` user:

```
make install &&
install -v -m755 -d /usr/share/doc/gutenprint-5.3.4/api/gutenprint{,ui2} &&
install -v -m644 doc/gutenprint/html/* \
          /usr/share/doc/gutenprint-5.3.4/api/gutenprint &&
install -v -m644 doc/gutenprintui2/html/* \
          /usr/share/doc/gutenprint-5.3.4/api/gutenprintui2
```

Command Explanations

`sed -i '....' ...Makefile.in`: This command is used so that the package documentation is installed in the conventional `/usr/share/doc` directory structure instead of `/usr/share/gutenprint/doc`.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--without-gimp2`: This switch disables support for GIMP because it has not been updated to support GIMP3 yet.

Configuring Gutenprint

Configuration Information

For CUPS to see newly installed print drivers, it has to be restarted (as the `root` user):

```
systemctl restart cups
```

Then point your web browser to <http://localhost:631/> to add a new printer to CUPS.

Contents

Installed Programs: cups-calibrate, cups-genppd.5.2, cups-genppdupdate, escputil, and testpattern

Installed Libraries: libgutenprint.so, libgutenprintui2.so and optionally, various CUPS filters and backend drivers under `/usr/lib/gutenprint/5.2/modules/`

Installed Directories: `/usr/{include,lib,share}/gutenprint`, `/usr/include/gutenprintui2` and `/usr/share/doc/gutenprint-5.3.4`

Short Descriptions

<code>cups-calibrate</code>	calibrates the color output of printers using the Gutenprint , CUPS or ESP Print Pro drivers
<code>cups-genppd.5.2</code>	generates Gutenprint PPD files for use with CUPS
<code>cups-genppdupdate</code>	regenerates the Gutenprint PPD files in use by CUPS
<code>escputil</code>	is a command line utility to perform various maintenance tasks on Epson Stylus inkjet printers
<code>testpattern</code>	is a test program to learn how to use libgutenprint

libcupsfilters-2.0.0

Introduction to libcupsfilters

The libcupsfilters library contains filter functions for the data format conversion tasks needed in Printer Applications. It also contains several API functions for developing printer drivers/Printer Applications.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/OpenPrinting/libcupsfilters/releases/download/2.0.0/libcupsfilters-2.0.0.tar.xz>
- Download MD5 sum: 267e569145bd20615fa18ae65ea6f870
- Download size: 1.2 MB
- Estimated disk space required: 38 MB (with tests)
- Estimated build time: 0.1 SBU (Using parallelism=4; with tests)

libcupsfilters Dependencies

Required

[Cups-2.4.10](#), [GLib-2.80.4](#), [ghostscript-10.03.1](#) or [mupdf-1.24.8](#), [Little CMS-2.16](#), [Poppler-24.08.0](#), and [Qpdf-11.9.1](#)

Recommended

[Dejavu fonts](#) (the build fails if they are not installed, although they are only required for the tests), [libexif-0.6.24](#), [libjpeg-turbo-3.0.1](#), [libpng-1.6.43](#), and [libtiff-4.6.0](#)

Installation of libcupsfilters

Install libcupsfilters by running the following commands:

```
./configure --prefix=/usr \
            --disable-static \
            --docdir=/usr/share/doc/libcupsfilters-2.0.0 &&
make
```

To test the results, issue `make check`. DejaVu fonts are needed for the tests.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-test-font-path=VALUE`: If you wish to run the tests, but you do not have the default `/usr/share/fonts/dejavu/DejaVuSans.ttf` font installed, use this switch to specify where `DejaVuSans.ttf` (or perhaps some other text TTF font - though this is untested) is located.

Contents

Installed Programs: None

Installed Libraries: `libcupsfilters.so`

Installed Directories: `/usr/include/cupsfilters`, `/usr/share/cups/{banners,charsets,data}`, and `/usr/share/doc/libcupsfilters-2.0.0`

Short Descriptions

`libcupsfilters.so` contains CUPS filters API functions

libppd-2.0.0

Introduction to libppd

The libppd library contains functions for handling legacy printers using PPD description files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/OpenPrinting/libppd/releases/download/2.0.0/libppd-2.0.0.tar.xz>
- Download MD5 sum: ab889cbcdaa23689d9a425769f208da9
- Download size: 567 KB
- Estimated disk space required: 13 MB (with tests)
- Estimated build time: less than 0.1 SBU (Using parallelism=4; with tests)

libppd Dependencies

Required

[libcupsfilters-2.0.0](#)

Installation of libppd

Install libppd by running the following commands:

```
./configure --prefix=/usr          \
            --disable-static      \
            --with-cups-rundir=/run/cups \
            --enable-ppdc-utils    \
            --docdir=/usr/share/doc/libppd-2.0.0 &&
make
```

To test the results, issue `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--with-cups-rundir=/run/cups`: Defines the directory containing transient printing data. The default `/var/run/cups` is obsolete.

`--enable-ppdc-utils`: Allows building utilities to manipulate `.ppd` files.

Contents

Installed Programs: ppdc, ppdhtml, ppdi, ppdmerge, and ppdo

Installed Libraries: libppd.so

Installed Directories: /usr/include/ppd, /usr/share/ppdc, and /usr/share/doc/libppd-2.0.0

Short Descriptions

<code>ppdc</code>	compiles PPDC source files into one or more PPD files
<code>ppdhtml</code>	reads a driver information file and produces a HTML summary page that lists all of the drivers in a file and the supported options
<code>ppdi</code>	imports one or more PPD files into a PPD compiler source file
<code>ppdmerge</code>	merges two or more PPD files into a single, multi-language PPD file
<code>ppdo</code>	extracts UI strings from PPDC source files and updates either a GNU gettext or Mac OS X strings format message catalog source file for translation
<code>libppd.so</code>	contains API functions for manipulating <code>.ppd</code> files

Chapter 47. Scanning

This chapter contains scanning applications which allow you to convert printed documents into formatted documents readable by other applications.

SANE-1.2.1

Introduction to SANE

SANE is short for Scanner Access Now Easy. Scanner access, however, is far from easy, since every vendor has their own protocols. The only known protocol that should bring some unity into this chaos is the TWAIN interface, but this is too imprecise to allow a stable scanning framework. Therefore, SANE comes with its own protocol, and the vendor drivers can't be used.

SANE is split into back ends and front ends. The back ends are drivers for the supported scanners and cameras. The front ends are user interfaces to access the backends.

This package is known to build and work properly using an LFS 12.2 platform.

Back Ends Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/sane/sane-backends-1.2.1.tar.gz>
- Download MD5 sum: 302159419ed1ee216c6e1edbe97c2a8c
- Download size: 6.9 MB
- Estimated disk space required: 311 MB (with tests)
- Estimated build time: 0.7 SBU (Using parallelism=4; with tests)

Note

The upstream location for this file is at <https://gitlab.com/sane-project> but the file URL uses an embedded md5sum. For convenience, the package has been copied to the location above.

SANE Dependencies

Optional

[Avahi-0.8](#), [Cups-2.4.10](#), [libjpeg-turbo-3.0.1](#), [libtiff-4.6.0](#), [libusb-1.0.27](#), [v4l-utils-1.28.1](#), [Net-SNMP](#), [libieee1284](#), [libgphoto2](#), and [texlive-20240312](#) (or [install-tl-unx](#))

Kernel Configuration, Drivers and Additional Packages

To access your scanner, you will probably need the related kernel drivers and/or additional support packages. A SCSI scanner will need SCSI drivers, a parallel port scanner needs parallel port support (you should use enhanced EPP modes) and perhaps [libieee1284](#), and a USB scanner will need the [libusb-1.0.27](#) package. For HP devices, you may need [hplip](#). Ensure you have the necessary drivers properly configured to access the devices.

Installation of SANE

Installation of SANE Backends

The SANE daemon should run with its own group. Create this group by issuing the following command as the `root` user:

```
groupadd -g 70 scanner
```

Note

`configure` assumes that the user building SANE Backends is a member of the `scanner` group for testing the existence of this group. As a result, the user building the package needs to be added to the `scanner` group. Add the user to the `scanner` group by issuing the following command as the `root` user:

```
usermod -G scanner -a username
```

After adding the `scanner` group as above, you will need to log out and back in to activate it.

For a USB scanner, if you are linking to [libusb-1.0.27](#), include the configure switch `--enable-libusb_1_0`. Install SANE Backends by running the following commands:

```
sg scanner -c "                                \
PYTHON=python3 ./configure --prefix=/usr      \
--sysconfdir=/etc                            \
--localstatedir=/var                          \
--with-lockdir=/run/lock \
--docdir=/usr/share/doc/sane-backends-1.2.1" &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install                               &&
install -m 644 -v tools/udev/libsane.rules   \
          /etc/udev/rules.d/65-scanner.rules &&
mkdir -p          /run/lock/sane &&
chgrp -v scanner /run/lock/sane
```

With the scanner on, run `scainfo` and the name and location of the device should appear. Of course, you need the device drivers configured, in order to run this test.

Command Explanations

`sg scanner -c "...":` runs the command between double quotes with principal group `scanner`. This may be needed if the user has been newly added to group `scanner`, and has not logged out and in.

`PYTHON=python3:` This switch forces the build system to use Python 3 instead of Python 2.

`--sysconfdir=/etc:` This switch installs the configuration files in `/etc/sane.d` instead of `/usr/etc/sane.d`.

Configuring SANE

Config Files

`/etc/sane.d/*.conf`

Configuration Information

Backend Configuration

The backend configuration files are located in `/etc/sane.d`. Information for configuring the various backends can be found by using the `man(5)` page for the desired backend. Run `man sane-<backend>`, substituting the desired backend.

Add any users that will use a scanner to the `scanner` group.

If you want to access a network scanner, include two lines in `net.conf`, as the `root` user (make sure to replace `<server_ip>` by the actual value):

```
cat >> /etc/sane.d/net.conf << "EOF"
connect_timeout = 60
<server_ip>
EOF
```

On the server side, include the client ip in the access list of `/etc/sane.d/saned.conf`, restart the `saned` daemon, and make sure the firewall, if any, is open to the client.

General Information

For general information about configuring and using SANE, see `man sane`. For information about USB scanning devices, run `man sane-usb`. For information about SCSI devices, run `man sane-scsi`.

Configuration and setup of the 'saned' daemon

The `saned` daemon is not meant to be used for untrusted clients. You should provide [Firewalling](#) protection to ensure only trusted clients access the daemon. Due to the complex security requirements to ensure only trusted clients access the daemon, BLFS does not provide instructions to configure the `saned` daemon. Some good (but outdated) information for setting up and securing the `saned` daemon can be found at <https://penguin-breeder.org/sane/saned/>.

Contents

Installed Programs: `gamma4scainfo`, `sane-config`, `saned`, `sane-find-scanner`, `scainfo`, and `umax_pp`

Installed Libraries: `libsane.so` and numerous scanner backend modules

Installed Directories: `/etc/sane.d`, `/usr/{include,lib,share}/sane`, and `/usr/share/doc/sane-1.2.1`

Short Descriptions

<code>gamma4scainfo</code>	creates a gamma table in the format expected by <code>scainfo</code>
<code>sane-config</code>	is a tool used to determine the compiler and linker flags that should be used to compile and link SANE
<code>saned</code>	is the SANE daemon that allows remote clients to access image acquisition devices available on the local host
<code>sane-find-scanner</code>	is a command-line tool to find SCSI and USB scanners and determine their device files. Its primary purpose is to make sure that scanners can be detected by SANE backends

<code>scanimate</code>	is a command line interface for scanning from image acquisition devices such as flatbed scanners or cameras. It is also used to list the available backend devices
<code>umax_pp</code>	is a tool used to read information from UMAX parallel port scanners. It can also be used for performing diagnostics on these scanners
<code>libsane.so</code>	is the application programming interface that is used to communicate between frontends and backends
<code>libsane-* .so</code>	modules are backend scanning library plugins used to interface with scanning devices. See http://www.sane-project.org/sane-supported-devices.html for a list of supported backends

Simple-scan-46.0

Introduction to Simple-scan

The Simple-scan package contains a modern document scanning application. It captures images using SANE, and supports the latest color management and graphics standards.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://download.gnome.org/sources/simple-scan/46/simple-scan-46.0.tar.xz>
- Download MD5 sum: 5ae9ca4d4c59520736c11a2eb427502d
- Download size: 1.1 MB
- Estimated disk space required: 27 MB
- Estimated build time: 0.1 SBU

Simple-scan Dependencies

Required

[colord-1.4.7](#), [itstool-2.0.7](#), [libgusb-0.4.9](#), [libadwaita-1.5.3](#), [libwebp-1.4.0](#), and [SANE-1.2.1](#)

Installation of Simple-scan

Install Simple-scan by running the following commands:

```
mkdir build &&
cd build &&

meson setup --prefix=/usr --buildtype=release .. &&
ninja
```

This package does not come with a test suite.

Now, as the `root` user:

```
ninja install &&
gtk-update-icon-cache -qtf /usr/share/icons/hicolor
```

Command Explanations

`--buildtype=release`: Specify a buildtype suitable for stable releases of the package, as the default may produce unoptimized binaries.

`gtk-update-icon-cache [...]`: This command updates the system-wide GTK+ icon cache since Simple-scan's build system does not update it to account for the icons that it installs.

Contents

Installed Programs: simple-scan

Installed Libraries: None

Installed Directories: /usr/share/help/*/simple-scan

Short Descriptions

Chapter 48. Standard Generalized Markup Language (SGML)

This chapter contains DocBook SGML document type definitions (DTDs), DocBook DSSSL Stylesheets and DocBook tools to validate, transform, format and publish DocBook documents.

sgml-common-0.6.3

Introduction to SGML Common

The SGML Common package contains `install-catalog`. This is useful for creating and maintaining centralized SGML catalogs.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sourceware.org/ftp/docbook-tools/new-trials/SOURCES/sgml-common-0.6.3.tgz>
- Download MD5 sum: 103c9828f24820df86e55e7862e28974
- Download size: 75 KB
- Estimated disk space required: 1.5 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/sgml-common-0.6.3-manpage-1.patch>

Installation of SGML Common

Instead of the normal convention of including the autotools files in the package, the maintainers included symlinks to the files in `/usr/share/automake`. For previous versions of Automake this convention is correct, but recent versions of Automake install the internal files in version specific directories. This causes the `configure` script to abort. To fix this error, the autotools are regenerated. Since the included `Makefile.am` file uses a syntax not supported by current versions of Automake, a patch is required to fix the syntax.

```
patch -Np1 -i ../sgml-common-0.6.3-manpage-1.patch &&
autoreconf -f -i
```

Install SGML Common by running the following commands:

```
./configure --prefix=/usr --sysconfdir=/etc &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make docdir=/usr/share/doc install &&

install-catalog --add /etc/sgml/sgml-ent.cat \
/usr/share/sgml/sgml-iso-entities-8879.1986/catalog &&

install-catalog --add /etc/sgml/sgml-docbook.cat \
/etc/sgml/sgml-ent.cat
```

Update Hint

Remove the above catalog items prior to upgrading (as the `root` user) with:

```
install-catalog --remove /etc/sgml/sgml-ent.cat \
/usr/share/sgml/sgml-iso-entities-8879.1986/catalog &&
```

```
install-catalog --remove /etc/sgml/sgml-docbook.cat \
/etc/sgml/sgml-ent.cat
```

Configuring SGML Common

Config Files

/etc/sgml/sgml.conf

Configuration Information

No change in this file is necessary.

Contents

Installed Programs: install-catalog and sgmlwhich

Installed Libraries: None

Installed Files: SGML and XML DocBook entity files

Installed Directories: /etc/sgml, /usr/share/doc/sgml-common-0.6.3, and /usr/share/sgml

Short Descriptions

install-catalog	creates a centralized catalog that maintains references to catalogs scattered throughout the /usr/share/sgml directory tree
sgmlwhich	prints to standard output the name of the main configuration file
SGML entities files	contains the basic character entities defined with SDATA entries
XML entities files	contains the basic character entities defined by a hexadecimal representation of the Unicode character number

docbook-3.1-dtd

Introduction to DocBook-3.1 SGML DTD

The DocBook SGML DTD package contains document type definitions for verification of SGML data files against the DocBook rule set. These are useful for structuring books and software documentation to a standard allowing you to utilize transformations already written for that standard.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.docbook.org/sgml/3.1/docbk31.zip>
- Download MD5 sum: 432749c0c806dbae81c8bcb70da3b5d3
- Download size: 55 KB
- Estimated disk space required: 676 KB
- Estimated build time: less than 0.1 SBU

DocBook-3.1 SGML DTD Dependencies

Required

[sgml-common-0.6.3](#) and [UnZip-6.0](#) (or [libarchive-3.7.4](#))

Installation of DocBook-3.1 SGML DTD

Note

The package source is distributed in `zip` format and requires `unzip` (or `bsdunzip` from libarchive). You should create a directory and change to that directory before unzipping the file to ease the removal of the source files after the package has been installed.

Install DocBook-3.1 SGML DTD by running the following commands:

```
sed -i -e '/ISO 8879/d' \
-e 's|DTDDECL "-//OASIS//DTD DocBook V3.1//EN"|"SGMLDECL|g' \
docbook.cat
```

This package does not come with a test suite.

Now, as the `root` user:

```
install -v -d -m755 /usr/share/sgml/docbook/sgml-dtd-3.1 &&
chown -R root:root . &&
install -v docbook.cat /usr/share/sgml/docbook/sgml-dtd-3.1/catalog &&
cp -v -af *.dtd *.mod *.dcl /usr/share/sgml/docbook/sgml-dtd-3.1 &&

install-catalog --add /etc/sgml/sgml-docbook-dtd-3.1.cat \
/usr/share/sgml/docbook/sgml-dtd-3.1/catalog &&

install-catalog --add /etc/sgml/sgml-docbook-dtd-3.1.cat \
/etc/sgml/sgml-docbook.cat
```

Command Explanations

`sed -i -e '/ISO 8879/d' docbook.cat`: This command removes the ENT definitions from the catalog file.

`sed -i -e 's|DTDDECL "-//OASIS//DTD DocBook V3.1//EN"|"SGMLDECL|g' docbook.cat`: This command replaces the DTDDECL catalog entry, which is not supported by Linux SGML tools, with the SGMLDECL catalog entry.

Configuring DocBook-3.1 SGML DTD

Config Files

`/etc/sgml/catalog`

Configuration Information

The above installation script updates the catalog.

Using only the most current 3.x version of DocBook SGML DTD requires the following (perform as the `root` user):

```
cat >> /usr/share/sgml/docbook/sgml-dtd-3.1/catalog << "EOF"
-- Begin Single Major Version catalog changes --

PUBLIC "-//Davenport//DTD DocBook V3.0//EN" "docbook.dtd"

-- End Single Major Version catalog changes --
EOF
```

Contents

Installed Programs: None

Installed Libraries: None

Installed Files: SGML DTD and MOD files

Installed Directory: `/usr/share/sgml/docbook/sgml-dtd-3.1`

Short Descriptions

SGML DTD files	contains a document type definition which defines the element types and the attribute lists that can be used in the corresponding SGML files
SGML MOD files	contains components of the document type definition that are sourced into the DTD files

docbook-4.5-dtd

Introduction to DocBook-4.5 SGML DTD

The DocBook-4.5 SGML DTD package contains document type definitions for verification of SGML data files against the DocBook rule set. These are useful for structuring books and software documentation to a standard allowing you to utilize transformations already written for that standard.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.docbook.org/sgml/4.5/docbook-4.5.zip>
- Download MD5 sum: 07c581f4bbcba6d3aac85360a19f95f
- Download size: 70 KB
- Estimated disk space required: 784 KB
- Estimated build time: less than 0.1 SBU

DocBook-4.5 SGML DTD Dependencies

Required

[sgml-common-0.6.3](#) and [UnZip-6.0](#) (or [libarchive-3.7.4](#))

Installation of DocBook-4.5 SGML DTD

Note

The package source is distributed in `zip` format and requires `unzip` (or `bsdunzip` from libarchive). You should create a directory and change to that directory before unzipping the file to ease the removal of the source files after the package has been installed.

Install DocBook-4.5 SGML DTD by running the following commands:

```
sed -i -e '/ISO 8879/d' \
-e '/gml/d' docbook.cat
```

This package does not come with a test suite.

Now, as the `root` user:

```
install -v -d /usr/share/sgml/docbook/sgml-dtd-4.5 &&
chown -R root:root . &&

install -v docbook.cat /usr/share/sgml/docbook/sgml-dtd-4.5/catalog &&
cp -v -af *.dtd *.mod *.dcl /usr/share/sgml/docbook/sgml-dtd-4.5 &&

install-catalog --add /etc/sgml/sgml-docbook-dtd-4.5.cat \
/usr/share/sgml/docbook/sgml-dtd-4.5/catalog &&

install-catalog --add /etc/sgml/sgml-docbook-dtd-4.5.cat \
/etc/sgml/sgml-docbook.cat
```

Command Explanations

`sed -i -e '/ISO 8879/d' -e '/gml/d' docbook.cat`: This command removes the ENT definitions from the catalog file.

Configuring DocBook-4.5 SGML DTD

Config Files

`/etc/sgml/catalog`

Configuration Information

The above installation script updates the catalog.

Using only the most current 4.x version of DocBook SGML DTD requires the following (perform as the `root` user):

```

cat >> /usr/share/sgml/docbook/sgml-dtd-4.5/catalog << "EOF"
-- Begin Single Major Version catalog changes --

PUBLIC "-//OASIS//DTD DocBook V4.4//EN" "docbook.dtd"
PUBLIC "-//OASIS//DTD DocBook V4.3//EN" "docbook.dtd"
PUBLIC "-//OASIS//DTD DocBook V4.2//EN" "docbook.dtd"
PUBLIC "-//OASIS//DTD DocBook V4.1//EN" "docbook.dtd"
PUBLIC "-//OASIS//DTD DocBook V4.0//EN" "docbook.dtd"

-- End Single Major Version catalog changes --
EOF

```

Contents

Installed Programs: None

Installed Libraries: None

Installed Files: SGML DTD and MOD files

Installed Directory: /usr/share/sgml/docbook/sgml-dtd-4.5

Short Descriptions

SGML DTD files	contains a document type definition which defines the element types and the attribute lists that can be used in the corresponding SGML files
SGML MOD files	contains components of the document type definition that are sourced into the DTD files

OpenSP-1.5.2

Introduction to OpenSP

The OpenSP package contains a C++ library for using SGML/XML files. This is useful for validating, parsing and manipulating SGML and XML documents.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/openjade/OpenSP-1.5.2.tar.gz>
- Download MD5 sum: 670b223c5d12cee40c9137be86b6c39b
- Download size: 1.5 MB
- Estimated disk space required: 32 MB
- Estimated build time: 1.0 SBU

Additional Downloads

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/OpenSP-1.5.2-gcc14-1.patch>

OpenSP Dependencies

Required

[sgml-common-0.6.3](#)

Optional

[libns1-2.0.1](#) and [xmlto-0.0.29](#)

Installation of OpenSP

Install OpenSP by running the following commands:

```

patch -Np1 -i ./OpenSP-1.5.2-gcc14-1.patch      &&
sed -i 's/32,/253,' lib/Syntax.cxx              &&
sed -i 's/LITLEN        240 /LITLEN          8092/' \
      unicode/{gensyntax.pl,unicode.syn}          &&

```

```

./configure --prefix=/usr \
--disable-static \
--disable-doc-build \
--enable-default-catalog=/etc/sgml/catalog \
--enable-http \
--enable-default-search-path=/usr/share/sgml &&

make pkgdatadir=/usr/share/sgml/OpenSP-1.5.2

```

To test the results, issue: `make check`. As many as nine of the 23 tests may fail. Do not be alarmed.

Now, as the `root` user:

```

make pkgdatadir=/usr/share/sgml/OpenSP-1.5.2 \
docdir=/usr/share/doc/OpenSP-1.5.2 \
install &&

ln -v -sf onsgmls /usr/bin/nsgmls &&
ln -v -sf osgmlnorm /usr/bin/sgmlnorm &&
ln -v -sf ospam /usr/bin/spam &&
ln -v -sf ospcat /usr/bin/spcat &&
ln -v -sf ospent /usr/bin/spent &&
ln -v -sf osx /usr/bin/sx &&
ln -v -sf osx /usr/bin/sgml2xml &&
ln -v -sf libosp.so /usr/lib/libsp.so

```

Command Explanations

`sed -i 's/32,/253,/...unicode.syn}`: These sed's prevent some annoying messages that may otherwise appear while running `openjade`.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-http`: This switch adds support for HTTP.

`--disable-doc-build`: This switch prevents the `configure` script checking if you have `xmldoc` installed. If you have `xmldoc`, you can remove this option.

`--enable-default-catalog=/etc/sgml/catalog`: This switch sets the path to the centralized catalog.

`--enable-default-search-path`: This switch sets the default value of `SGML_SEARCH_PATH`.

`--enable-xml-messages`: This switch adds support for XML Formatted Messages.

`make pkgdatadir=/usr/share/sgml/OpenSP-1.5.2` : This sets the `pkgdatadir` variable in the `Makefile` from `/usr/share/OpenSP` to `/usr/share/sgml/OpenSP-1.5.2`.

`ln -v -sf ...`: These commands create the SP equivalents of OpenSP executables and libraries.

Contents

Installed Programs: `onsgmls`, `osgmlnorm`, `ospam`, `ospcat`, `ospent`, `osx`, and the SP equivalent symlinks: `nsgmls`, `sgml2xml`, `sgmlnorm`, `spam`, `spcat`, `spent`, and `sx`

Installed Library: `libosp.so` and the SP equivalent symlink: `libsp.so`

Installed Directories: `/usr/include/OpenSP`, `/usr/share/doc/OpenSP`, and `/usr/share/sgml/OpenSP-1.5.2`

Short Descriptions

<code>onsgmls</code>	is used to process SGML files
<code>osgmlnorm</code>	prints on the standard output a normalized document instance for the SGML document contained in the concatenation of the entities with system identifiers <code>.nf</code> and <code>.fi</code>
<code>ospam</code>	is a markup stream editor
<code>ospcat</code>	prints effective system identifiers found in the catalogs
<code>ospent</code>	provides access to OpenSP's entity manager
<code>osx</code>	is an SGML normalizer or used to convert SGML files to XML files
<code>nsgmls</code>	is a symlink to <code>onsgmls</code>
<code>sgml2xml</code>	is a symlink to <code>osx</code>

```

sgmlnorm      is a symlink to osgmlnorm
spam          is a symlink to ospam
spcat         is a symlink to ospcat
spent         is a symlink to ospent
sx            is a symlink to osx
libosp.so     contains functions required by the OpenSP programs to parse, validate and manipulate SGML and XML files
libosp.so     is a symlink to libosp.so

```

OpenJade-1.3.2

Introduction to OpenJade

The OpenJade package contains a DSSSL engine. This is useful for SGML and XML transformations into RTF, TeX, SGML and XML.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/openjade/openjade-1.3.2.tar.gz>
- Download MD5 sum: 7df692e3186109cc00db6825b777201e
- Download size: 880 KB
- Estimated disk space required: 19.2 MB
- Estimated build time: 0.7 SBU

Additional Download

- Required patch: <https://www.linuxfromscratch.org/patches/blfs/12.2/openjade-1.3.2-upstream-1.patch>

OpenJade Dependencies

Required

[OpenSP-1.5.2](#)

Installation of OpenJade

First fix problems when building with newer compilers:

```
patch -Np1 -i ../openjade-1.3.2-upstream-1.patch
```

Now fix a compilation problem with perl-5.16 and later:

```
sed -i -e '/getopts/{N;s#\&G#g;s#do .getopts.pl.;##;}' \
-e '/use POSIX/ause Getopt::Std;' msggen.pl
```

Install OpenJade by running the following commands:

```

export CXXFLAGS="${CXXFLAGS:--O2 -g} -fno-lifetime-dse"           &&
./configure --prefix=/usr                                         \
--mandir=/usr/share/man                                       \
--enable-http                                                 \
--disable-static                                              \
--enable-default-catalog=/etc/sgml/catalog \
--enable-default-search-path=/usr/share/sgml \
--datadir=/usr/share/sgml/openjade-1.3.2    &&
make

```

This package does not come with a test suite.

Now, as the `root` user:

```

make install
make install-man
ln -v -sf openjade /usr/bin/jade
ln -v -sf libogrove.so /usr/lib/libgrove.so
ln -v -sf libospgrove.so /usr/lib/libspgrove.so
ln -v -sf libostyle.so /usr/lib/libstyle.so

install -v -m644 dsssl/catalog /usr/share/sgml/openjade-1.3.2/ &&
install -v -m644 dsssl/*.{dtd,dsl,sgm} \
    /usr/share/sgml/openjade-1.3.2 &&
install-catalog --add /etc/sgml/openjade-1.3.2.cat \
    /usr/share/sgml/openjade-1.3.2/catalog &&
install-catalog --add /etc/sgml/sgml-docbook.cat \
    /etc/sgml/openjade-1.3.2.cat

```

Command Explanations

`export CXXFLAGS=...:` This command prevents segmentation faults when the package is compiled with gcc-6.1.

`make install-man:` This command installs the `openjade` man page.

`--disable-static:` This switch prevents the building of the static library.

`--enable-http:` This switch adds support for HTTP.

`--enable-default-catalog=/etc/sgml/catalog:` This switch sets the path to the centralized catalog.

`--enable-default-search-path:` This switch sets the default value of `SGML_SEARCH_PATH`.

`--datadir=/usr/share/sgml/openjade-1.3.2:` This switch puts data files in `/usr/share/sgml/openjade-1.3.2` instead of `/usr/share`.

`ln -v -sf ...:` These commands create the Jade equivalents of OpenJade executables and libraries.

Configuring OpenJade

Configuration Information

As the `root` user:

```

echo "SYSTEM \"http://www.oasis-open.org/docbook/xml/4.5/docbookx.dtd\" \
\"/usr/share/xml/docbook/xml-dtd-4.5/docbookx.dtd\"" >> \
/usr/share/sgml/openjade-1.3.2/catalog

```

This configuration is only necessary if you intend to use OpenJade to process the BLFS XML files through DSSSL Stylesheets.

Contents

Installed Programs: `openjade` and the Jade equivalent symlink, `jade`

Installed Libraries: `libogrove.so`, `libospgrove.so`, `libostyle.so`, and the Jade equivalent symlinks: `libgrove.so`, `libspgrove.so`, and `libstyle.so`

Installed Directory: `/usr/share/sgml/openjade-1.3.2`

Short Descriptions

<code>openjade</code>	is a DSSSL engine used for transformations
<code>jade</code>	is a symlink to <code>openjade</code>

docbook-dsssl-1.79

Introduction to DocBook DSSSL Stylesheets

The DocBook DSSSL Stylesheets package contains DSSSL stylesheets. These are used by OpenJade or other tools to transform SGML and XML DocBook files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/docbook/docbook-dsssl-1.79.tar.bz2>
- Download (FTP):
- Download MD5 sum: bc192d23266b9a664ca0aba4a7794c7c
- Download size: 277 KB
- Estimated disk space required: 14 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

Documentation and test data

- Download (HTTP): <https://downloads.sourceforge.net/docbook/docbook-dsssl-doc-1.79.tar.bz2>
- Download MD5 sum: 9a7b809a21ab7d2749bb328334c380f2
- Download size: 142 KB

DocBook DSSSL Stylesheets Dependencies

Required

[sgml-common-0.6.3](#)

Required (to Test the DocBook SGML Toolchain)

[docbook-3.1-dtd](#), [docbook-4.5-dtd](#), [OpenSP-1.5.2](#), and [OpenJade-1.3.2](#)

Installation of DocBook DSSSL Stylesheets

If you downloaded the documentation, run:

```
tar -xf ../docbook-dsssl-doc-1.79.tar.bz2 --strip-components=1
```

Install DocBook DSSSL Stylesheets by running the following commands as the `root` user:

```
install -v -m755 bin/collateindex.pl /usr/bin          &&
install -v -m644 bin/collateindex.pl.1 /usr/share/man/man1    &&
install -v -d -m755 /usr/share/sgml/docbook/dsssl-stylesheets-1.79 &&
cp -v -R * /usr/share/sgml/docbook/dsssl-stylesheets-1.79      &&

install-catalog --add /etc/sgml/dsssl-docbook-stylesheets.cat \
    /usr/share/sgml/docbook/dsssl-stylesheets-1.79/catalog      &&

install-catalog --add /etc/sgml/dsssl-docbook-stylesheets.cat \
    /usr/share/sgml/docbook/dsssl-stylesheets-1.79/common/catalog &&

install-catalog --add /etc/sgml/sgml-docbook.cat           \
    /etc/sgml/dsssl-docbook-stylesheets.cat
```

Command Explanations

The above commands create an installation script for this package.

Testing the DocBook SGML Toolchain (Optional)

The following commands will perform the necessary tests to confirm that your installed DocBook SGML toolchain will produce desired results. You must have the [docbook-3.1-dtd](#), [docbook-4.5-dtd](#), [OpenSP-1.5.2](#), and [OpenJade-1.3.2](#) packages installed, and perform the tests as the `root` user.

All tests will be performed from the `/usr/share/sgml/docbook/dsssl-stylesheets-1.79/doc/testdata` directory as the `root` user:

```
cd /usr/share/sgml/docbook/dsssl-stylesheets-1.79/doc/testdata
```

The first test should produce no output to stdout (your screen) and create a file named `jtest.rtf` in the current directory:

```
openjade -t rtf -d jtest.dsl jtest.sgm
```

The next test should return only the following line to stdout: onsgmls:I: "OpenSP" version "1.5.2"

```
onsgmls -sv test.sgm
```

The next test should produce no output to stdout and create a file named `test.rtf` in the current directory:

```
openjade -t rtf \  
-d /usr/share/sgml/docbook/dsssl-stylesheets-1.79/print/docbook.dsl \  
test.sgm
```

The last test should produce no output to stdout and create a file named `c1.htm` in the current directory:

```
openjade -t sgml \  
-d /usr/share/sgml/docbook/dsssl-stylesheets-1.79/html/docbook.dsl \  
test.sgm
```

Finally, clean up:

```
rm jtest.rtf test.rtf c1.htm
```

Contents

Installed Program: `collateindex.pl`

Installed Libraries: None

Installed Files: DSSSL stylesheets

Installed Directory: `/usr/share/sgml/docbook/dsssl-stylesheets-1.79`

Short Descriptions

`collateindex.pl` is a Perl script that creates a DocBook index from raw index data

DocBook-utils-0.6.14

Introduction to DocBook-utils

The DocBook-utils package is a collection of utility scripts used to convert and analyze SGML documents in general, and DocBook files in particular. The scripts are used to convert from DocBook or other SGML formats into “classical” file formats like HTML, man, info, RTF and many more. There’s also a utility to compare two SGML files and only display the differences in markup. This is useful for comparing documents prepared for different languages.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://sourceware.org/ftp/docbook-tools/new-trials/SOURCES/docbook-utils-0.6.14.tar.gz>
- Download MD5 sum: 6b41b18c365c01f225bc417cf632d81c
- Download size: 124 KB
- Estimated disk space required: 1.44 MB
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/docbook-utils-0.6.14-grep_fix-1.patch

DocBook-utils Dependencies

Required

[OpenJade-1.3.2](#), [docbook-dsssl-1.79](#), and [docbook-3.1-dtd](#)

Recommended (Runtime Dependencies Only)

Installation of DocBook-utils

Install DocBook-utils by running the following commands:

```
patch -Np1 -i ../docbook-utils-0.6.14-grep_fix-1.patch &&
sed -i 's:/html::' doc/HTML/Makefile.in &&
./configure --prefix=/usr --mandir=/usr/share/man &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make docdir=/usr/share/doc install
```

Many packages use an alternate name for the DocBook-utils scripts. If you wish to create these alternate names, use the following command as the `root` user:

```
for doctype in html ps dvi man pdf rtf tex texi txt
do
  ln -svf docbook2$doctype /usr/bin/db2$doctype
done
```

Note

The `jw` script uses the `which` command to locate required utilities. You must install [Which-2.21](#) before attempting to use any of the DocBook-utils programs.

Command Explanations

`patch -Np1 -i ../docbook-utils-0.6.14-grep_fix-1.patch`: This patch corrects the syntax in the `jw` (Jade Wrapper) script which is at the heart of much `db2*` processing, so that the current version of grep will not reject it.

`sed -i 's:/html::' doc/HTML/Makefile.in`: This command changes the installation directory of the HTML documents.

`docdir=/usr/share/doc`: This option is placed on the `make install` line because it is not recognized by `configure`.

Contents

Installed Programs: `docbook2dvi`, `docbook2html`, `docbook2man`, `docbook2pdf`, `docbook2ps`, `docbook2rtf`, `docbook2tex`, `docbook2texi`, `docbook2txt`, `jw`, and `sgmldiff`

Installed Libraries: None

Installed Directories: `/usr/share/doc/docbook-utils-0.6.14` and `/usr/share/sgml/docbook/utils-0.6.14`

Installed Symlinks: `db2dvi`, `db2html`, `db2man`, `db2pdf`, `db2ps`, `db2rtf`, `db2tex`, `db2texi`, and `db2txt`

Short Descriptions

<code>docbook2*</code>	are simple one-line wrapper scripts to <code>jw</code> . They are provided as easy-to-remember names used to convert DocBook or other SGML files to the respective format
<code>db2*</code>	are symlinks pointing at the respectively named <code>docbook2*</code> commands, created to satisfy some program's use of these names
<code>jw</code>	is a script used to convert DocBook or other SGML files to various output formats. It hides most of OpenJade's complexity and adds comfortable features
<code>sgmldiff</code>	is used to compare two SGML files and only return the differences in the markup. This is especially useful to compare files that should be identical except for language differences in the content

Chapter 49. Extensible Markup Language (XML)

This chapter contains the DocBook XML document type definition (DTD) and DocBook Stylesheets which are used to validate, transform, format and publish DocBook documents.

docbook-xml-4.5

Introduction to DocBook-4.5 XML DTD

The DocBook-4.5 XML DTD-4.5 package contains document type definitions for verification of XML data files against the DocBook rule set. These are useful for structuring books and software documentation to a standard allowing you to utilize transformations already written for that standard.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.docbook.org/xml/4.5/docbook-xml-4.5.zip>
- Download MD5 sum: 03083e288e87a7e829e437358da7ef9e
- Download size: 96 KB
- Estimated disk space required: 1.2 MB
- Estimated build time: less than 0.1 SBU

DocBook-4.5 XML DTD Dependencies

Required

[libxml2-2.13.3](#) and [UnZip-6.0](#) (or [libarchive-3.7.4](#))

Installation of DocBook-4.5 XML DTD

Note

The package source is distributed in `zip` format and requires `unzip` (or `bsdunzip` from libarchive). You should create a directory and change to that directory before unzipping the file to ease the removal of the source files after the package has been installed.

Install DocBook XML DTD by running the following commands as the `root` user:

```
install -v -d -m755 /usr/share/xml/docbook/xml-dtd-4.5 &&
install -v -d -m755 /etc/xml &&
cp -v -af --no-preserve=ownership docbook.cat *.dtd ent/*.*.mod \
    /usr/share/xml/docbook/xml-dtd-4.5
```

Create (or update) and populate the `/etc/xml/docbook` catalog file by running the following commands as the `root` user:

```
if [ ! -e /etc/xml/docbook ]; then
    xmllcatalog --noout --create /etc/xml/docbook
fi &&
xmllcatalog --noout --add "public" \
    "--//OASIS//DTD DocBook XML V4.5//EN" \
    "http://www.oasis-open.org/docbook/xml/4.5/docbookx.dtd" \
    /etc/xml/docbook &&
xmllcatalog --noout --add "public" \
    "--//OASIS//DTD DocBook XML CALS Table Model V4.5//EN" \
    "file:///usr/share/xml/docbook/xml-dtd-4.5/calstblx.dtd" \
    /etc/xml/docbook &&
xmllcatalog --noout --add "public" \
    "--//OASIS//DTD XML Exchange Table Model 19990315//EN" \
    "file:///usr/share/xml/docbook/xml-dtd-4.5/soextblx.dtd" \
    /etc/xml/docbook &&
xmllcatalog --noout --add "public" \
    "--//OASIS//ELEMENTS DocBook XML Information Pool V4.5//EN" \
    "file:///usr/share/xml/docbook/xml-dtd-4.5/dbpoolx.mod" \
    /etc/xml/docbook &&
xmllcatalog --noout --add "public" \
    "--//OASIS//ELEMENTS DocBook XML Document Hierarchy V4.5//EN" \
    "file:///usr/share/xml/docbook/xml-dtd-4.5/dbhierx.mod" \
    /etc/xml/docbook &&
xmllcatalog --noout --add "public" \
    "--//OASIS//ELEMENTS DocBook XML HTML Tables V4.5//EN" \
```

```

"file:///usr/share/xml/docbook/xml-dtd-4.5/htmltblx.mod" \
/etc/xml/docbook &&
xmllcatalog --noout --add "public" \
"--//OASIS//ENTITIES DocBook XML Notations V4.5//EN" \
"file:///usr/share/xml/docbook/xml-dtd-4.5/dbnotnx.mod" \
/etc/xml/docbook &&
xmllcatalog --noout --add "public" \
"--//OASIS//ENTITIES DocBook XML Character Entities V4.5//EN" \
"file:///usr/share/xml/docbook/xml-dtd-4.5/dbcentx.mod" \
/etc/xml/docbook &&
xmllcatalog --noout --add "public" \
"--//OASIS//ENTITIES DocBook XML Additional General Entities V4.5//EN" \
"file:///usr/share/xml/docbook/xml-dtd-4.5/dbgenent.mod" \
/etc/xml/docbook &&
xmllcatalog --noout --add "rewriteSystem" \
"http://www.oasis-open.org/docbook/xml/4.5" \
"file:///usr/share/xml/docbook/xml-dtd-4.5" \
/etc/xml/docbook &&
xmllcatalog --noout --add "rewriteURI" \
"http://www.oasis-open.org/docbook/xml/4.5" \
"file:///usr/share/xml/docbook/xml-dtd-4.5" \
/etc/xml/docbook

```

Create (or update) and populate the `/etc/xml/catalog` catalog file by running the following commands as the `root` user:

```

if [ ! -e /etc/xml/catalog ]; then
    xmllcatalog --noout --create /etc/xml/catalog
fi &&
xmllcatalog --noout --add "delegatePublic" \
"--//OASIS//ENTITIES DocBook XML" \
"file:///etc/xml/docbook" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegatePublic" \
"--//DTD DocBook XML" \
"file:///etc/xml/docbook" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateSystem" \
"http://www.oasis-open.org/docbook/" \
"file:///etc/xml/docbook" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://www.oasis-open.org/docbook/" \
"file:///etc/xml/docbook" \
/etc/xml/catalog

```

Configuring DocBook-4.5 XML DTD

Config Files

`/etc/xml/catalog` and `/etc/xml/docbook`

Configuration Information

Caution

Various BLFS packages request DocBook XML DTD version 4.x before V4.5, so the following step must be done for those packages to be built successfully.

The above installation creates the files and updates the catalogs. In order to utilize DocBook XML DTD V4.5 when any version 4.x is requested in the System Identifier, you need to add additional statements to the catalog files. If you have any of the DocBook XML DTD's referenced below already installed on your system, remove those entries from the `for` command below (issue the commands as the `root` user):

```

for DTDVERSION in 4.1.2 4.2 4.3 4.4
do
    xmllcatalog --noout --add "public" \
"--//OASIS//DTD DocBook XML V$DTDVERSION//EN" \
"http://www.oasis-open.org/docbook/xml/$DTDVERSION/docbookx.dtd" \
/etc/xml/docbook
    xmllcatalog --noout --add "rewriteSystem" \

```

```

"http://www.oasis-open.org/docbook/xml/$DTDVERSION" \
"file:///usr/share/xml/docbook/xml-dtd-4.5" \
/etc/xml/docbook
xmlcatalog --noout --add "rewriteURI" \
"http://www.oasis-open.org/docbook/xml/$DTDVERSION" \
"file:///usr/share/xml/docbook/xml-dtd-4.5" \
/etc/xml/docbook
xmlcatalog --noout --add "delegateSystem" \
"http://www.oasis-open.org/docbook/xml/$DTDVERSION/" \
"file:///etc/xml/docbook" \
/etc/xml/catalog
xmlcatalog --noout --add "delegateURI" \
"http://www.oasis-open.org/docbook/xml/$DTDVERSION/" \
"file:///etc/xml/docbook" \
/etc/xml/catalog
done

```

Contents

Installed Programs: None

Installed Libraries: None

Installed Files: DTD, MOD and ENT files

Installed Directories: /etc/xml and /usr/share/xml/docbook/xml-dtd-4.5

Short Descriptions

DTD files	contain a document type definition which defines the element types and the attribute lists that can be used in the corresponding XML files
MOD files	contain components of the document type definition that are sourced into the DTD files
ENT files	contain lists of named character entities allowed in HTML

docbook-xml-5.0

Introduction to DocBook XML DTD and Schemas 5.0

The DocBook XML DTD and Schemas-5.0 package contains document type definitions and schemas for verification of XML data files against the DocBook rule set. These are useful for structuring books and software documentation to a standard allowing you to utilize transformations already written for that standard. In addition to providing a DTD, version 5 introduced the RelaxNG schema and Schematron rules, and is incompatible with previous versions of DocBook XML.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://docbook.org/xml/5.0/docbook-5.0.zip>
- Download MD5 sum: 2411c19ed4fb141f3fa3d389fae40736
- Download size: 820 KB
- Estimated disk space required: 6.2 MB
- Estimated build time: 0.1 SBU

DocBook XML DTD and Schemas 5.0 Dependencies

Required

[libxml2-2.13.3](#) and [UnZip-6.0](#)

Installation of DocBook XML DTD and Schemas 5.0

Install DocBook XML DTD and Schemas by running the following commands as the `root` user:

```

install -vdm755 /usr/share/xml/docbook/schema/{dtd,rng,sch,xsd}/5.0 &&
install -vm644 dtd/* /usr/share/xml/docbook/schema/dtd/5.0           &&
install -vm644 rng/* /usr/share/xml/docbook/schema/rng/5.0            &&

```

```
install -vm644 sch/* /usr/share/xml/docbook/schema/sch/5.0      &&
install -vm644 xsd/* /usr/share/xml/docbook/schema/xsd/5.0
```

Create (or update) and populate the `/etc/xml/docbook-5.0` catalog file by running the following commands as the `root` user:

```
if [ ! -e /etc/xml/docbook-5.0 ]; then
    xmllcatalog --noout --create /etc/xml/docbook-5.0
fi &&

xmllcatalog --noout --add "public" \
"-//OASIS//DTD DocBook XML 5.0//EN" \
"file:///usr/share/xml/docbook/schema/dtd/5.0/docbook.dtd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "system" \
"http://www.oasis-open.org/docbook/xml/5.0/dtd/docbook.dtd" \
"file:///usr/share/xml/docbook/schema/dtd/5.0/docbook.dtd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "system" \
"http://docbook.org/xml/5.0/dtd/docbook.dtd" \
"file:///usr/share/xml/docbook/schema/dtd/5.0/docbook.dtd" \
/etc/xml/docbook-5.0 &&

xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rng/docbook.rng" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbook.rng" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbook.rng" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbook.rng" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rng/docbookxi.rng" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbookxi.rng" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbookxi.rng" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbookxi.rng" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rnc/docbook.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbook.rnc" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbook.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbook.rnc" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rnc/docbookxi.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbookxi.rnc" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbookxi.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.0/docbookxi.rnc" \
/etc/xml/docbook-5.0 &&

xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/docbook.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/docbook.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/docbook.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/docbook.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/docbookxi.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/docbookxi.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/docbookxi.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/docbookxi.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/xi.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/xi.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
```

```

"http://docbook.org/xml/5.0/xsd/xi.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/xi.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/xlink.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/xlink.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/xlink.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/xlink.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/xml.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/xml.xsd" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/xml.xsd" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/xml.xsd" \
/etc/xml/docbook-5.0 &&

xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/sch/docbook.sch" \
"file:///usr/share/xml/docbook/schema/sch/5.0/docbook.sch" \
/etc/xml/docbook-5.0 &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/sch/docbook.sch" \
"file:///usr/share/xml/docbook/schema/sch/5.0/docbook.sch" \
/etc/xml/docbook-5.0

```

While again as the `root` user, create the individual catalogs:

```

xmllcatalog --noout --create /usr/share/xml/docbook/schema/dtd/5.0/catalog.xml &&

xmllcatalog --noout --add "public" \
"--//OASIS//DTD DocBook XML 5.0//EN" \
"docbook.dtd" /usr/share/xml/docbook/schema/dtd/5.0/catalog.xml &&
xmllcatalog --noout --add "system" \
"http://www.oasis-open.org/docbook/xml/5.0/dtd/docbook.dtd" \
"docbook.dtd" /usr/share/xml/docbook/schema/dtd/5.0/catalog.xml &&

xmllcatalog --noout --create /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbook.rng" \
"docbook.rng" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rng/docbook.rng" \
"docbook.rng" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbookxi.rng" \
"docbookxi.rng" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rng/docbookxi.rng" \
"docbookxi.rng" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbook.rnc" \
"docbook.rnc" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rng/docbook.rnc" \
"docbook.rnc" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/rng/docbookxi.rnc" \
"docbookxi.rnc" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/rng/docbookxi.rnc" \
"docbookxi.rnc" /usr/share/xml/docbook/schema/rng/5.0/catalog.xml &&

xmllcatalog --noout --create /usr/share/xml/docbook/schema/sch/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/sch/docbook.sch" \
"docbook.sch" /usr/share/xml/docbook/schema/sch/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/sch/docbook.sch" \
"docbook.sch" /usr/share/xml/docbook/schema/sch/5.0/catalog.xml &&

xmllcatalog --noout --create /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \

```

```

"http://docbook.org/xml/5.0/xsd/docbook.xsd" \
"docbook.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/docbook.xsd" \
"docbook.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/docbookxi.xsd" \
"docbookxi.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/docbookxi.xsd" \
"docbookxi.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/xlink.xsd" \
"xlink.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/xlink.xsd" \
"xlink.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://docbook.org/xml/5.0/xsd/xml.xsd" \
"xml.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml &&
xmllcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.0/xsd/xml.xsd" \
"xml.xsd" /usr/share/xml/docbook/schema/xsd/5.0/catalog.xml

```

Create (or update) and populate the system XML catalog (`/etc/xml/catalog`) by running the following commands as the `root` user:

```

if [ ! -e /etc/xml/catalog ]; then
    xmllcatalog --noout --create /etc/xml/catalog
fi &&
xmllcatalog --noout --add "delegatePublic" \
"-//OASIS//DTD DocBook XML 5.0//EN" \
"file:///usr/share/xml/docbook/schema/dtd/5.0/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateSystem" \
"http://docbook.org/xml/5.0/dtd/" \
"file:///usr/share/xml/docbook/schema/dtd/5.0/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.0/dtd/" \
"file:///usr/share/xml/docbook/schema/dtd/5.0/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.0/rng/" \
"file:///usr/share/xml/docbook/schema/rng/5.0/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.0/sch/" \
"file:///usr/share/xml/docbook/schema/sch/5.0/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.0/xsd/" \
"file:///usr/share/xml/docbook/schema/xsd/5.0/catalog.xml" \
/etc/xml/catalog

```

Configuring DocBook XML DTD and Schemas 5.0

Config Files

`/etc/xml/catalog` and `/etc/xml/docbook-5.0`

Contents

Installed Files: DTD, RNG, SCH and XSD files

Installed Directories: `/etc/xml` and `/usr/share/xml/docbook/schema/{dtd,rng,sch,xsd}/5.0`

Short Descriptions

DTD files	contain the DocBook 5.0 document type definitions which define the element types and the attribute lists that can be used in the corresponding XML files
RNC, RNG, and SCH files	contain the Docbook 5.0 RelaxNG, RelaxNG Compact, and Schematron schema definitions

docbook-xml-5.1

Introduction to DocBook XML Schemas 5.1

The DocBook XML Schemas-5.1 package contains schema files and Schematron rules for verification of XML data files against the DocBook rule set. These are useful for structuring books and software documentation to a standard allowing you to utilize transformations already written for that standard.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://docbook.org/xml/5.1/docbook-v5.1-os.zip>
- Download MD5 sum: d8bea8ddfc5743578a31cb18f9ae1f5a
- Download size: 752 KB
- Estimated disk space required: 8.5 MB
- Estimated build time: less than 0.1 SBU

DocBook XML Schemas 5.1 Dependencies

Required

[libxml2-2.13.3](#) and [UnZip-6.0](#) (or [libarchive-3.7.4](#))

Installation of DocBook XML Schemas 5.1

Note

The package source is distributed in `zip` format and requires `unzip` (or `bsdunzip` from libarchive). You should create a directory and change to that directory before unzipping the file to ease the removal of the source files after the package has been installed.

Install DocBook XML Schemas by running the following commands as the `root` user:

```
install -vdm755 /usr/share/xml/docbook/schema/{rng,sch}/5.1      &&
install -m644   schemas/rng/* /usr/share/xml/docbook/schema/rng/5.1 &&
install -m644   schemas/sch/* /usr/share/xml/docbook/schema/sch/5.1 &&
install -m755   tools/db4-entities.pl /usr/bin                  &&
install -vdm755 /usr/share/xml/docbook/stylesheet/docbook5      &&
install -m644   tools/db4-upgrade.xsl \
               /usr/share/xml/docbook/stylesheet/docbook5
```

Create (or update) and populate the `/etc/xml/docbook-5.1` catalog file by running the following commands as the `root` user:

```
if [ ! -e /etc/xml/docbook-5.1 ]; then
    xmllcatalog --noout --create /etc/xml/docbook-5.1
fi &&

xmllcatalog --noout --add "uri" \
    "http://www.oasis-open.org/docbook/xml/5.1/rng/docbook.rng" \
    "file:///usr/share/xml/docbook/schema/rng/5.1/docbook.rng" \
    /etc/xml/docbook-5.1 &&
xmllcatalog --noout --add "uri" \
    "http://docbook.org/xml/5.1/rng/docbook.rng" \
    "file:///usr/share/xml/docbook/schema/rng/5.1/docbook.rng" \
    /etc/xml/docbook-5.1 &&
xmllcatalog --noout --add "uri" \
    "http://www.oasis-open.org/docbook/xml/5.1/rng/docbookxi.rng" \
    "file:///usr/share/xml/docbook/schema/rng/5.1/docbookxi.rng" \
    /etc/xml/docbook-5.1 &&
xmllcatalog --noout --add "uri" \
    "http://docbook.org/xml/5.1/rng/docbookxi.rng" \
    "file:///usr/share/xml/docbook/schema/rng/5.1/docbookxi.rng" \
    /etc/xml/docbook-5.1 &&
```

```

xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/rnc/docbook.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.1/docbook.rnc" \
/etc/xml/docbook-5.1 &&
xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/rng/docbook.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.1/docbook.rnc" \
/etc/xml/docbook-5.1 &&
xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/rnc/docbookxi.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.1/docbookxi.rnc" \
/etc/xml/docbook-5.1 &&
xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/rng/docbookxi.rnc" \
"file:///usr/share/xml/docbook/schema/rng/5.1/docbookxi.rnc" \
/etc/xml/docbook-5.1 &&

xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/sch/docbook.sch" \
"file:///usr/share/xml/docbook/schema/sch/5.1/docbook.sch" \
/etc/xml/docbook-5.1 &&
xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/sch/docbook.sch" \
"file:///usr/share/xml/docbook/schema/sch/5.1/docbook.sch" \
/etc/xml/docbook-5.1

```

While again as the `root` user, create the individual catalogs:

```

xmlcatalog --noout --create /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&

xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/schemas/rng/docbook.schemas/rng" \
"docbook.schemas/rng" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/schemas/rng/docbook.schemas/rng" \
"docbook.schemas/rng" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/schemas/rng/docbookxi.schemas/rng" \
"docbookxi.schemas/rng" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/schemas/rng/docbookxi.schemas/rng" \
"docbookxi.schemas/rng" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/schemas/rng/docbook.rnc" \
"docbook.rnc" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/schemas/rng/docbook.rnc" \
"docbook.rnc" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/schemas/rng/docbookxi.rnc" \
"docbookxi.rnc" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/schemas/rng/docbookxi.rnc" \
"docbookxi.rnc" /usr/share/xml/docbook/schema/rng/5.1/catalog.xml &&
xmlcatalog --noout --create /usr/share/xml/docbook/schema/sch/5.1/catalog.xml &&

xmlcatalog --noout --add "uri" \
"http://docbook.org/xml/5.1/schemas/sch/docbook.schemas/sch" \
"docbook.schemas/sch" /usr/share/xml/docbook/schema/sch/5.1/catalog.xml &&
xmlcatalog --noout --add "uri" \
"http://www.oasis-open.org/docbook/xml/5.1/schemas/sch/docbook.schemas/sch" \
"docbook.schemas/sch" /usr/share/xml/docbook/schema/sch/5.1/catalog.xml

```

Create (or update) and populate the system XML catalog (`/etc/xml/catalog`) by running the following commands as the `root` user:

```

if [ ! -e /etc/xml/catalog ]; then
  xmlcatalog --noout --create /etc/xml/catalog
fi &&
xmlcatalog --noout --add "delegatePublic" \
"--//OASIS//DTD DocBook XML 5.1//EN" \
"file:///usr/share/xml/docbook/schema/dtd/5.1/catalog.xml" \
/etc/xml/catalog &&
xmlcatalog --noout --add "delegateSystem" \
"http://docbook.org/xml/5.1/dtd/" \

```

```

"file:///usr/share/xml/docbook/schema/dtd/5.1/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.1/dtd/" \
"file:///usr/share/xml/docbook/schema/dtd/5.1/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.1/rng/" \
"file:///usr/share/xml/docbook/schema/rng/5.1/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.1/sch/" \
"file:///usr/share/xml/docbook/schema/sch/5.1/catalog.xml" \
/etc/xml/catalog &&
xmllcatalog --noout --add "delegateURI" \
"http://docbook.org/xml/5.1/xsd/" \
"file:///usr/share/xml/docbook/schema/xsd/5.1/catalog.xml" \
/etc/xml/catalog

```

Configuring DocBook XML Schemas 5.1

Config Files

/etc/xml/catalog and /etc/xml/docbook-5.1

Contents

Installed Programs: db4-entities.pl

Installed Files: RNC, RNG, SCH, and XSL files

Installed Directories: /etc/xml, /usr/share/xml/docbook/stylesheets, and /usr/share/xml/docbook/schema/{rng,sch}/5.1

Short Descriptions

db4-entities.pl	is a script to convert DocBook 4.x files to DocBook 5.1
RNC, RNG, and SCH files	contain the Docbook 5.1 RelaxNG, RelaxNG Compact, and Schematron schema definitions
db4-upgrade.xsl	is an XSL stylesheet to assist in conversion from DocBook 4.x files to DocBook 5.1

docbook-xsl-nons-1.79.2

Introduction to DocBook XSL Stylesheets

The DocBook XSL Stylesheets package contains XSL stylesheets. These are useful for performing transformations on XML DocBook files.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/docbook/xslt10-stylesheets/releases/download/release/1.79.2/docbook-xsl-nons-1.79.2.tar.bz2>
- Download MD5 sum: 2666d1488d6ced1551d15f31d7ed8c38
- Download size: 22 MB
- Estimated disk space required: 58 MB (includes installing optional documentation)
- Estimated build time: less than 0.1 SBU

Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/docbook-xsl-nons-1.79.2-stack_fix-1.patch

Optional documentation

- Download (HTTP): <https://github.com/docbook/xslt10-stylesheets/releases/download/release/1.79.2/docbook-xsl-doc-1.79.2.tar.bz2>
- Download MD5 sum: 62375ca864fc198cb2b17d98209d0b8c

- Download size: 522 KB

DocBook XSL Stylesheets Dependencies

Recommended (at runtime)

[libxml2-2.13.3](#)

Optional (all used at runtime)

[apache-ant-1.10.14](#) (to produce “webhelp” documents), [libxslt-1.1.42](#) (or any other XSLT processor), to process Docbook documents, [Ruby-3.3.4](#) (to utilize the “epub” stylesheets), [Zip-3.0](#) (to produce “epub3” documents), and [Saxon6](#) and [Xerces2 Java](#) (used with [apache-ant-1.10.14](#) to produce “webhelp” documents)

Installation of DocBook XSL Stylesheets

First, fix a problem that causes stack overflows when doing recursion:

```
patch -Np1 -i ../docbook-xsl-nons-1.79.2-stack_fix-1.patch
```

If you downloaded the optional documentation tarball, unpack it with the following command:

```
tar -xf ../docbook-xsl-doc-1.79.2.tar.bz2 --strip-components=1
```

BLFS does not install the required packages to run the test suite and provide meaningful results.

Install DocBook XSL Stylesheets by running the following commands as the `root` user:

```
install -v -m755 -d /usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2 &&
cp -v -R VERSION assembly common eclipse epub epub3 extensions fo \
    highlighting html htmlhelp images javahelp lib manpages params \
    profiling roundtrip slides template tests tools webhelp website \
    xhtml xhtml1_1 xhtml5 \
/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2 &&
ln -s VERSION /usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2/VERSION.xsl &&
install -v -m644 -D README \
    /usr/share/doc/docbook-xsl-nons-1.79.2/README.txt &&
install -v -m644 RELEASE-NOTES* NEWS* \
    /usr/share/doc/docbook-xsl-nons-1.79.2
```

If you downloaded the optional documentation tarball, install the documentation by issuing the following command as the `root` user:

```
cp -v -R doc/* /usr/share/doc/docbook-xsl-nons-1.79.2
```

Configuring DocBook XSL Stylesheets

Config Files

/etc/xml/catalog

Configuration Information

Note

If you are installing the current version of docbook-xsl-nons over a previous version of docbook-xsl, then remove the old rewrite entries in the catalog as the `root` user:

```
sed -i '/rewrite/d' /etc/xml/catalog
```

Create (or append) and populate the XML catalog file using the following commands as the `root` user (both `http` and `https` forms are used because upstream have had both in their documentation):

```

if [ ! -d /etc/xml ]; then install -v -m755 -d /etc/xml; fi &&
if [ ! -f /etc/xml/catalog ]; then
    xmllcatalog --noout --create /etc/xml/catalog
fi &&

xmllcatalog --noout --add "rewriteSystem" \
    "http://cdn.docbook.org/release/xsl-nons/1.79.2" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteSystem" \
    "https://cdn.docbook.org/release/xsl-nons/1.79.2" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteURI" \
    "http://cdn.docbook.org/release/xsl-nons/1.79.2" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteSystem" \
    "https://cdn.docbook.org/release/xsl-nons/current" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteSystem" \
    "https://cdn.docbook.org/release/xsl-nons/current" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteURI" \
    "http://cdn.docbook.org/release/xsl-nons/current" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteURI" \
    "https://cdn.docbook.org/release/xsl-nons/current" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteSystem" \
    "http://docbook.sourceforge.net/release/xsl/current" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteURI" \
    "http://docbook.sourceforge.net/release/xsl/current" \
    "/usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2" \
/etc/xml/catalog

```

Occasionally, you may find the need to install other versions of the XSL stylesheets as some projects reference a specific version. One example is BLFS-6.0, which required the 1.67.2 version. In these instances you should install any other required version in its own versioned directory and create catalog entries as follows (substitute the desired version number for `<version>`):

```

xmllcatalog --noout --add "rewriteSystem" \
    "http://docbook.sourceforge.net/release/xsl/<version>" \
    "/usr/share/xml/docbook/xsl-stylesheets-<version>" \
/etc/xml/catalog &&

xmllcatalog --noout --add "rewriteURI" \
    "http://docbook.sourceforge.net/release/xsl/<version>" \
    "/usr/share/xml/docbook/xsl-stylesheets-<version>" \
/etc/xml/catalog

```

Contents

Installed Files: /usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2/*/*.xsl

Installed Directories: /usr/share/xml/docbook/xsl-stylesheets-nons-1.79.2 and /usr/share/doc/docbook-xsl-nons-1.79.2

Short Descriptions

XSL Stylesheets are used for performing transformations on XML files

itstool-2.0.7

Introduction to itstool

Itstool extracts messages from XML files and outputs PO template files, then merges translations from MO files to create translated XML files. It determines what to translate and how to chunk it into messages using the W3C Internationalization Tag Set (ITS).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://files.itstool.org/itstool/itstool-2.0.7.tar.bz2>
- Download MD5 sum: 267a3bdc72a2d8abb1b824f2ea32ee9b
- Download size: 104 KB
- Estimated disk space required: 688 KB
- Estimated build time: less than 0.1 SBU

Itstool Dependencies

Required

[docbook-xml-4.5](#)

Installation of itstool

First, fix some compatibility problems with Python-3.12:

```
sed -i 's/re.sub(/re.sub(r'           itstool.in &&
sed -i 's/re.compile(/re.compile(r'  itstool.in
```

Install itstool by running the following commands:

```
PYTHON=/usr/bin/python3 ./configure --prefix=/usr &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: itstool

Installed Libraries: None

Installed Directory: /usr/share/itstool

Short Descriptions

itstool is used to create translated XML files

xmlio-0.0.29

Introduction to xmlio

The xmlio package is a front-end to a XSL toolchain. It chooses an appropriate stylesheet for the conversion you want and applies it using an external XSLT processor. It also performs any necessary post-processing.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://pagure.io/xmlto/archive/0.0.29/xmlto-0.0.29.tar.gz>
- Download MD5 sum: 556f2642cdcd005749bd4c08bc621c37
- Download size: 64 KB
- Estimated disk space required: 1.9 MB (with tests)
- Estimated build time: less than 0.1 SBU (with tests)

xmlto Dependencies

Required

[docbook-xml-4.5](#), [docbook-xsl-nons-1.79.2](#), and [libxslt-1.1.42](#)

Optional (for DVI, PDF, and postscript backend post-processing)

[fop-2.9](#), [dblateX](#), and [PassiveTeX](#)

Optional (for text backend post-processing)

One of [Links-2.30](#), [Lynx-2.9.2](#), [W3m](#), or [Elinks](#)

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/xmlto>

Installation of xmlto

Install xmlto by running the following commands:

```
autoreconf -fiv &&
LINKS="/usr/bin/links" ./configure --prefix=/usr &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`LINKS="/usr/bin/links"`: This environment variable fixes a bug causing `xmlto` to think that `links` command is the same as `elinks`. Setting this variable does not cause problems if `links` is not installed, unless you have `Elinks` installed, and wish to use it for text backend post-processing, in which case, remove it.

Contents

Installed Programs: `xmllif` and `xmlto`

Installed Libraries: None

Installed Directory: `/usr/share/xmlto`

Short Descriptions

`xmllif` is a conditional processing instructions for XML
`xmlto` applies an XSL stylesheet to an XML document

Chapter 50. PostScript

This chapter includes applications that create, manipulate or view PostScript files and create or view Portable Document Format PDF files.

Enscript-1.6.6

Introduction to Enscript

Enscript converts ASCII text files to PostScript, HTML, RTF, ANSI and overstrikes.

This package is known to build and work properly using an LFS 12.2 platform.

Caution

Enscript cannot convert UTF-8 encoded text to PostScript. The issue is discussed in detail in the [Needed Encoding Not a Valid Option](#) section of the [Locale Related Issues](#) page. The solution is to use [paps-0.8.0](#), instead of Enscript, for converting UTF-8 encoded text to PostScript.

Package Information

- Download (HTTP): <https://ftp.gnu.org/gnu/enscript/enscript-1.6.6.tar.gz>
- Download MD5 sum: 3acc242b829adacabcaf28533f049afdf
- Download size: 1.3 MB
- Estimated disk space required: 14 MB
- Estimated build time: 0.1 SBU

Enscript Dependencies

Optional

[texlive-20240312](#) (or [install-tl-unx](#))

Installation of Enscript

Install Enscript by running the following commands:

```
./configure --prefix=/usr          \
            --sysconfdir=/etc/enscript \
            --localstatedir=/var        \
            --with-media=Letter &&
make &&
pushd docs &&
makeinfo --plaintext -o enscript.txt enscript.texi &&
popd
```

If you have [texlive-20240312](#) installed, you can create Postscript and PDF documentation by issuing (does not support parallel make):

```
make -j1 -C docs ps pdf
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install &&
install -v -m755 -d /usr/share/doc/enscript-1.6.6 &&
install -v -m644 README* *.txt docs/*.txt \
/usr/share/doc/enscript-1.6.6
```

If you built Postscript and PDF documentation, install it using the following command as the `root` user:

```
install -v -m644 docs/*.{dvi,ps,ps} \
/usr/share/doc/enscript-1.6.6
```

Command Explanations

`--sysconfdir=/etc/enscript`: This switch puts configuration data in `/etc/enscript` instead of `/usr/etc`.

`--localstatedir=/var`: This switch sets the directory for runtime data to `/var` instead of `/usr/var`.

--with-media=Letter: This switch sets the medium format to letter size instead of the A4 default.

Contents

Installed Programs: diffpp, enscript, mkafmmap, over, sliceprint, and states

Installed Libraries: None

Installed Directories: /etc/enscript, /usr/share/doc/enscript-1.6.6, and /usr/share/enscript

Short Descriptions

diffpp	converts <code>diff</code> output files to a format suitable to be printed with <code>enscript</code>
enscript	is a filter, used primarily by printing scripts, that converts ASCII text files to PostScript, HTML, RTF, ANSI and overstrikes
mkafmmap	creates a font map from a given file
over	is a script which calls <code>enscript</code> and passes the correct parameters to create overstriked fonts
sliceprint	slices documents with long lines
states	is an <code>awk</code> -like text processing tool with some state machine extensions. It is designed for program source code highlighting and for similar tasks where state information helps input processing

ePDFView-gtk3-20200814

Introduction to ePDFView-gtk3

ePDFView-gtk3 is a fork of the old ePDFView program. Although the github repository https://github.com/Flow-It/epdfview_old.git names it as 'old', it is the gtk3 fork. It is a lightweight replacement for Evince as it does not rely upon GNOME libraries and is more capable than MuPDF.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://anduin.linuxfromscratch.org/BLFS/epdfview-gtk3/epdfview-gtk3-20200814.tar.xz>
- Download MD5 sum: d222a3dc26c2faf6f862018bb478fb36
- Download size: 184 KB
- Estimated disk space required: 3.5 MB
- Estimated build time: less than 0.1 SBU (using parallelism=4)

ePDFView Dependencies

Required

[GTK+-3.24.43](#) and [Poppler-24.08.0](#)

Recommended

[Cups-2.4.10](#) (to access print queues), [desktop-file-utils-0.27](#), and [hicolor-icon-theme-0.18](#) (both for the icons this installs)

Optional

The home page of the project's repository mentions that there are optional dependencies, which are enabled by switches. These are for building the documentation and running the test suite, [Doxygen-1.12.0](#) and [Cppunit](#). Note that doxygen only installs a skeletal page about itself, which contains no package documentation, and the test code does not compile with recent versions of C++.

Installation of ePDFView-gtk3

Install ePDFView-gtk3 by running the following commands:

```
mkdir build &&
cd      build &&

meson setup --prefix=/usr           \
            --buildtype=release \
```

```
-D enable-printing=true \
..
&&
ninja
```

This package does not come with a buildable test suite.

Now, as the `root` user:

```
ninja install
```

Note

This package installs icon files into the `/usr/share/icons/hicolor` hierarchy and desktop files into the `/usr/share/applications` hierarchy. You can improve system performance and memory usage by updating `/usr/share/icons/hicolor/index.theme` and `/usr/share/applications/mimeinfo.cache`. To perform the update you must have [GTK+-3.24.43](#) installed (for the icon cache) and [desktop-file-utils-0.27](#) (for the desktop cache) and issue the following commands as the `root` user:

```
gtk-update-icon-cache -qtf /usr/share/icons/hicolor &&
update-desktop-database -q
```

Command Explanations

`-D enable-printing=true`: builds the code to link to the cups print queue(s). Omit this if you have not installed cups.

Configuring ePDFView-gtk3

ePDFView-gtk3 has several keyboard hotkeys for optional features. Most of them are 'off' by default, including the toolbar, and if it has been closed with the menu disabled, the program can start with all options not being visible until the relevant function keys are pressed.

- [F6] - toggle toolbar
- [F7] - toggle menu
- [F8] - toggle invert-colors
- [F9] - toggle show-index
- [F11] - toggle fullscreen

Contents

Installed Program: epdfview

Installed Libraries: None

Installed Directory: /usr/share/epdfview

Short Descriptions

`epdfview` is a Gtk+-3 program for viewing PDF documents

fop-2.9

Introduction to fop

The FOP (Formatting Objects Processor) package contains a print formatter driven by XSL formatting objects (XSL-FO). It is a Java application that reads a formatting object tree and renders the resulting pages to a specified output. Output formats currently supported include PDF, PCL, PostScript, SVG, XML (area tree representation), print, AWT, MIF and ASCII text. The primary output target is PDF.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://archive.apache.org/dist/xmlgraphics/fop/source/fop-2.9-src.tar.gz>
- Download MD5 sum: f7537ca7f2e16971fa99c8bb0dad62c7

- Download size: 20 MB
- Estimated disk space required: 333 MB (including files downloaded to the user directory)
- Estimated build time: 0.9 SBU

Additional Downloads

Required Additional Downloads:

- Maven build system:
<https://archive.apache.org/dist/maven/maven-3/3.9.4/binaries/apache-maven-3.9.4-bin.tar.gz>
0698a533397eda60cbebcc0fb68ae842
9.0 MB (additionally, about 90 MB are downloaded to the building user's directory)

Recommended packages

- Objects for Formatting Objects (OFFO) hyphenation patterns:
<https://downloads.sourceforge.net/offo/2.2/offo-hyphenation.zip>
bf9c09bf05108ef9661b8f08d91c2336
862 KB

fop Dependencies

Required

[apache-ant-1.10.14](#)

Optional

[a graphical environment](#) (to run tests), [JAI Image I/O Tools](#), and [JEuclid](#)

Installation of fop

Ensure \$JAVA_HOME is set correctly before beginning the build. To build the JIMI SDK and/or XMLUnit extension classes, ensure the corresponding .jar files can be found via the CLASSPATH environment variable.

Installing OFFO Hyphenation Patterns

Copy the XML hyphenation patterns into the fop source tree by running the following commands:

```
unzip ..\offo-hyphenation.zip &&
cp offo-hyphenation/hypn/* fop/hypn &&
rm -rf offo-hyphenation
```

Installing a temporary Maven binary

Starting with fop-2.5, the Maven build system is required. We use the binary provided by apache, that we install in a temporary location:

```
tar -xf ..\apache-maven-3.9.4-bin.tar.gz -C /tmp
```

Installing fop Components

The javadoc command that ships with OpenJDK 10 and later has become much stricter than previous versions regarding conformance of the Javadoc comments in source code to HTML. The FOP documentation does not meet those standards, so the conformance checks have to be disabled. This can be done with the following command:

```
sed -i '\@</javadoc@i\
<arg value="-Xdoclint:none"/> \
<arg value="--allow-script-in-comments"/> \
<arg value="--ignore-source-errors"/>' \
fop/build.xml
```

Compile fop by running the following commands:

```
cd fop &&
LC_ALL=en_US.UTF-8 \
PATH=$PATH:/tmp/apache-maven-3.9.4/bin \
ant all javadocs &&
```

```
mv build/javadocs .
```

This package comes with a test suite, but the java infrastructure installed in this book does not allow running it.

Now, install Fop as the `root` user:

```
install -v -d -m755 -o root -g root      /opt/fop-2.9 &&
cp -vR build conf examples fop* javadocs lib /opt/fop-2.9 &&
chmod a+x /opt/fop-2.9/fop               &&
ln -v -sfn fop-2.9 /opt/fop
```

The last thing to do is to clean what we have done:

```
rm -rf /tmp/apache-maven-3.9.4
```

Command Explanations

`sed -i ... build.xml`: This adds three switches to the `javadoc` command, preventing some errors from occurring when building the documentation.

`export LC_ALL=en_US.UTF-8`: the compiler fails if using an ASCII locale.

`ant target`: This reads the file `build.xml` and builds the target: `compile` compiles the java sources, `jar-main` generates jar archives, `jar-hyphenation` generates the hyphenation patterns for FOP, `junit` runs the junit tests, and `javadoc` builds the documentation. The `all` target runs all of the above.

`ln -v -sfn fop-2.9 /opt/fop`: This is optional and creates a convenience symlink so that `$FOP_HOME` doesn't have to be changed each time there's a package version change.

Configuring fop

Config Files

`~/.foprc`

Configuration Information

Using fop to process some large FO's (including the FO derived from the BLFS XML sources), can lead to memory errors. Unless you add a parameter to the `java` command used in the `fop` script you may receive messages similar to the one shown below:

```
Exception in thread "main" java.lang.OutOfMemoryError: Java heap space
```

To avoid errors like this, you need to pass an extra parameter to the `java` command used in the `fop` script. This can be accomplished by creating a `~/.foprc` (which is sourced by the `fop` script) and adding the parameter to the `FOP_OPTS` environment variable.

The `fop` script looks for a `FOP_HOME` environment variable to locate the fop class libraries. You can create this variable using the `~/.foprc` file as well. Create a `~/.foprc` file using the following commands:

```
cat > ~/.foprc << "EOF"
FOP_OPTS="-Xmx<RAM_Installed>m"
FOP_HOME="/opt/fop"
EOF
```

Replace `<RAM_Installed>` with a number representing the amount of RAM installed in your computer (in megabytes). An example would be `FOP_OPTS="-Xmx768m"`.

To include the `fop` script in your path, update the system-wide profile with the following command as the `root` user:

```
cat > /etc/profile.d/fop.sh << "EOF"
# Begin /etc/profile.d/fop.sh

pathappend /opt/fop

# End /etc/profile.d/fop.sh
EOF
```

Note

Running `fop` can be somewhat verbose. The default logging level can be changed from INFO to any of FINEST, FINER, FINE, CONFIG, INFO, WARNING, SEVERE, ALL, or OFF. To do this, edit `$JAVA_HOME/jre/lib/logging.properties` and change the entries for `.level` and `java.util.logging.ConsoleHandler.level` to the desired value.

Contents

Installed Programs: fop

Installed Libraries: fop.jar and numerous support library classes located in `/opt/fop/{build,lib}`; JAI components include libmlib_jai.so, jai_codec.jar, jai_core.jar, and mlibwrapper_jai.jar

Installed Directory: /opt/fop-2.9

Short Descriptions

- `fop` is a wrapper script to the `java` command which sets up the `fop` environment and passes the required parameters
- `fop.jar` contains all the `fop` Java classes

MuPDF-1.24.8

Introduction to MuPDF

MuPDF is a lightweight PDF and XPS viewer.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://www.mupdf.com/downloads/archive/mupdf-1.24.8-source.tar.gz>
- Download MD5 sum: 68a444a4828626c0c8aa7552fef583bf
- Download size: 52 MB
- Estimated disk space required: 251 MB
- Estimated build time: 0.2 SBU (Using parallelism=4)

MuPDF Dependencies

Required

[GLU-9.0.3](#) and [Xorg Libraries](#)

Recommended

[harfBuzz-9.0.0](#), [libjpeg-turbo-3.0.1](#), [OpenJPEG-2.5.2](#), and [cURL-8.9.1](#)

Optional

[xdg-utils-1.2.1](#) (runtime), [jbig2dec](#), and [MuJS](#)

Required (runtime)

[a graphical environment](#)

Installation of MuPDF

Install MuPDF by running the following commands:

```
cat > user.make << EOF &&
USE_SYSTEM_FREETYPE := yes
USE_SYSTEM_HARFBUZZ := yes
USE_SYSTEM_JBIG2DEC := no
```

```

USE_SYSTEM_JPEGXR := no # not used without HAVE_JPEGXR
USE_SYSTEM_LCMS2 := no # need lcms2-art fork
USE_SYSTEM_LIBJPEG := yes
USE_SYSTEM_MUJS := no # build needs source anyway
USE_SYSTEM_OPENJPEG := yes
USE_SYSTEM_ZLIB := yes
USE_SYSTEM_GLUT := no # need freeglut2-art fork
USE_SYSTEM_CURL := yes
USE_SYSTEM_GUMBO := no
EOF

export XCFLAGS=-fPIC                                &&
make build=release shared=yes verbose=yes &&
unset XCFLAGS

```

This package does not come with a test suite.

Now, as the `root` user:

```

make prefix=/usr                                \
      shared=yes                                \
      docdir=/usr/share/doc/mupdf-1.24.8 \
      install                                     &&
ln -sfv libmupdf.so.24.8 /usr/lib/libmupdf.so   &&
ln -sfv libmupdf.so.24.8 /usr/lib/libmupdf.so.8  &&
chmod 755 /usr/lib/libmupdf.so.24.8            &&
ln -sfv mupdf-x11 /usr/bin/mupdf

```

Command Explanations

`ln -sfv mupdf-x11 /usr/bin/mupdf` : This symbolic link chooses between `mupdf-g1` and `mupdf-x11` when running `mupdf`.

Contents

Installed Program: `mupdf` (symlink), `mupdf-g1`, `mupdf-x11`, `mupdf-x11-curl`, `muraster`, and `mutool`

Installed Libraries: `libmupdf.so`

Installed Directories: `/usr/include/mupdf`, `/usr/share/doc/mupdf-1.24.8`

Short Descriptions

<code>mupdf</code>	is a program for viewing PDF, XPS, EPUB, and CBZ documents, and various image formats such as PNG, JPEG, GIFF, and TIFF
<code>mupdf-g1</code>	same as <code>mupdf</code> , using an opengl renderer
<code>mupdf-x11</code>	same as <code>mupdf</code> , using an X Window renderer
<code>muraster</code>	is a program used to perform rasterization tasks with PDF documents
<code>mutool</code>	is a program to perform various operations on PDF files, such as merging and cleaning PDF documents
<code>libmupdf.so</code>	contains the mupdf API functions

paps-0.8.0

Introduction to paps

paps is a text to PostScript converter that works through Pango. Its input is a UTF-8 encoded text file and it outputs vectorized PostScript. It may be used for printing any complex script supported by Pango.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/dov/paps/releases/download/v0.8.0/paps-0.8.0.tar.gz>
- Download MD5 sum: 6bd661b8fd224adc3343a91e6521a4f2
- Download size: 220 KB
- Estimated disk space required: 4.0 MB

- Estimated build time: less than 0.1 SBU

paps Dependencies

Required

[fmt-11.0.2](#) and [Pango-1.54.0](#)

Optional

[Doxygen-1.12.0](#)

Installation of paps

Install paps by running the following commands:

```
./configure --prefix=/usr      \
            --disable-Werror \
            --mandir=/usr/share/man &&
make
```

This package does not come with a test suite.

Now, as the `root` user:

```
make install
```

Contents

Installed Program: paps

Installed Library: None

Installed Directory: None

Short Descriptions

`paps` is a text to PostScript converter that supports UTF-8 character encoding

Chapter 51. Typesetting

This chapter includes texlive applications that create output equivalent to typesetting.

There are two alternative routes through this chapter:

Some people may wish to use the binary installer, either because of the complexity of installing all of texlive from source, or because they only want a part of the package, or perhaps because they wish to get updates as soon as those are available (the source is only updated annually, but the binaries and associated tex and sty files are updated for approximately 10 months). These people should begin at [Setting the PATH for TeX Live](#) and then follow the [install-tl-unx](#) instructions. After installing, they can run `tlmgr` to update their system.

Most people reading BLFS will wish to build from source. BLFS used to start by installing `install-tl-unx` and then use that to bootstrap the build. Nowadays, we build almost the whole of texlive *without* a binary install, by adding the separately-packaged texmf files into this build. For this, begin at [Setting the PATH for TeX Live](#) then go to [texlive-20240312](#) which will install most of texlive, together with *all* of the supporting files. This almost-complete install can then be used to build the remaining parts of texlive: [asymptote-2.91](#), [biber-2.20](#), [dvisvgm-3.4](#), and [xindy-2.5.1](#).

Because the texmf files (including documentation, fonts, scripts and supporting files) are all in one tarball, it is not practical to limit what gets extracted in a sensible way (you could exclude one or other of the typesetting engines, not its many supporting files) when building from source in this manner.

In either case, BLFS installs into `/opt/texlive/2024`.

Also, please note that texlive is released annually, and updating from the previous year to the current year is no longer supported. If for some reason you wish to keep versions for multiple years, for most things you can mount the old or new system at `/opt/texlive` and fix up your PATH as necessary. However doing that will not preserve any changes in `texmf-local` and if you build from source and try to run a newer version of biber with an older version of biblatex it is unlikely to work.

Setting the PATH for TeX Live

Upstream prefers to install in `/usr/local` but the BLFS editors regard that as inconvenient, and think using `/opt/texlive` is more appropriate. Originally BLFS used a full binary install to bootstrap the source install, so the same prefix is used for both.

Before starting to build TeX Live, set up your PATH so that the system can properly find the files. If you set up your login scripts as recommended in [The Bash Shell Startup Files](#), update the needed paths by creating the `texlive.sh` script. The programs are always installed in an `<ARCH>-linux` subdirectory and on 32-bit x86 this is always `i386-linux`. For `x86_64` and `i?86` we can generate this as `$TEXARCH`:

Note

If upgrading from a previous year's version, you should manually edit `texlive.sh` to ensure that the version for the year you wish to use is the only TeX present (some people need to keep multiple years available to ensure there are no regressions in their documents).

Now, create the `texlive.sh` script as the `root` user:

```
TEXARCH=$(uname -m | sed -e 's/i.86/i386/' -e 's/$/-linux/')

cat > /etc/profile.d/texlive.sh << EOF
# Begin texlive setup
TEXLIVE_PREFIX=/opt/texlive/2024
export TEXLIVE_PREFIX

pathappend \$TEXLIVE_PREFIX/texmf-dist/doc/info INFOPATH
pathappend \$TEXLIVE_PREFIX/bin/\$TEXARCH

TEXMFNF=\$TEXLIVE_PREFIX/texmf-dist/web2c
export TEXMFNF

# End texlive setup
EOF

unset TEXARCH
```

Note

The backslashes before the dollar signs in the script above are to facilitate a copy/paste operation. The backslashes should not appear in the actual script.

The new paths can be immediately activated by running:

```
source /etc/profile
```

You should now proceed either to [install-tl-ux](#) for a binary installation of texlive, or to [texlive-20240312](#) to begin installing from source.

install-tl-ux

Introduction to TeX Live and its installer

The TeX Live package is a comprehensive TeX document production system. It includes TeX, LaTeX2e, LuaLaTeX, Metafont, MetaPost, BibTeX and many other programs; an extensive collection of macros, fonts and documentation; and support for typesetting in many different scripts from around the world.

This page is for people who wish to use the binary installer to provide the programs, the scripts, and a lot of supporting files and documentation. The installer is updated frequently, so any published md5sum will soon be out of date. Newer versions of the installer are expected to work with these instructions, for so long as they install to a `2024/` directory.

There are two reasons why you may wish to install the binaries in BLFS: either you need a smaller install (e.g. at a minimum plain TeX without LaTeX, ConTeXt, etc), or you wish to use `tlmgr` to get updates whilst this version is supported (typically, until March of the year after it was released). For the latter, you might prefer to install in your `/home` directory as an unprivileged user, and to then make corresponding changes to the PATH in your `~/.bashrc` or equivalent.

Note

If you wish to use ConTeXt with `luametateX` (most of the old MKII and MKIV code was removed from TeX Live 2023 by the ConTeXt developer), using the binary is probably the easiest option. The source no-longer ships with TeX Live and is poorly adapted to building with systems except those running Mac and Windows. See comments 1 to 5 of [#17823](#).

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz>
- Download MD5 sum: Varies frequently
- Download size: 5.5 MB
- Estimated disk space required: 85 MB for plain TeX, typically 2 to 4 MB for latex and later engines, 8.6 GB if everything is included
- Estimated build time: varies, depending on network speed and traffic

Recommended

[GnuPG-2.4.5](#) (to validate both the initial downloads, and also any updates you might later make using `tlmgr`) and [libwww-perl-6.77](#) (to use a single connection to the server, which will reduce its load and speed things up)

Recommended (at runtime)

The binaries are mostly linked to included static libraries or general (LFS) system libraries, but a few of the programs and several scripts will fail if the following packages are not present:

[ghostscript-10.03.1](#) is dynamically loaded by the external application [dvisvgm](#), which is used by asy when that creates SVG files.

[Xorg Libraries](#) and [libxcb-1.17.0](#) are needed for inimf, mf, pdfclose, pdfopen and xdvi-xaw. But if you are using asy, or using a TeX engine to create a PDF file, you will need [a graphical environment](#) (for PDF files, this is to support a PDF viewer of your choice, for example [epdfview-gtk3-20200814](#)).

The binary version of asy needs [Freeglut-3.6.0](#).

The binary version of asy is linked to libGLX.so.0 from [libglvnd](#), but installing that will break future updates of BLFS packages such as [Mesa-24.1.5](#). Work around that by creating a symlink as the `root` user:

```
ln -sv libGL.so.1 /usr/lib/libGLX.so.0
```

The binary versions of biber and xindy are linked to `libcrypt.so.1` from old versions of glibc. To use these two applications, follow the Note about binary-only applications in libxcrypt in LFS to install the ABI version 1 crypt library.

Note

As always with contributed binary software, it is possible that the required dependencies may change when the installer is updated. In particular, these dependencies have only been checked on x86_64.

[Python2](#) is used by the unmaintained ebong CTAN module (intended for writing Bengali in Rapid Roman Format). `/usr/bin/python` is also in the shebang line for the latex-papersize and lilyglyphs scripts, and documentation at CTAN says both modules have been updated to work with python3. In pythontex there are scripts to invoke python3 or python2 according to the system's version of `python`. [Ruby-3.3.4](#) is used by two scripts, one is for pTeX (Japanese vertical writing) and the other is match_parens which might be generally useful. The perl module [Tk](#), which needs to be run from an X11 session to run the tests and requires [Tk-8.6.14](#) is used by one of the scripts for ptex and is needed for texdoctk (a GUI interface for finding documentation files and opening them with the appropriate viewer). `ps2pdf`, from [ghostscript-10.03.1](#), is used by some utilities and scripts.

Binary Installation of TeX Live

The TeX Live set of programs with its supporting documents, fonts, and utilities is very large. The upstream maintainers recommend placing all files in a single directory structure. BLFS recommends `/opt/texlive`.

Note

If you have chosen to install the binary as a normal user, the directory for the prefix needs to be writable by that user. The `root` user can chown `/opt/texlive/2024` to that user before the user starts the install. If any later change in that directory is made by the `root` user, *that will change the ownership*, which breaks usage by normal users.

As with any other package, unpack the installer and change into its directory, `install-tl-<CCYYMMDD>`. This directory name changes when the installer is updated, so replace `<CCYYMMDD>` by the correct directory name.

Note

The distribution binaries installed below may use static linking for general linux system libraries. Additional libraries or interpreters as specified in the dependencies section do not need to be present during the install, but the programs that need them will not run until their specific dependencies have been installed.

With all contributed binary software, there may be a mismatch between the builder's toolchain and your hardware. In most of TeX this will probably not matter, but in uncommon corner cases you might hit problems. For example, if your x86_64 processor does not support 3dnowext or 3dnow, the 2014-06-28 binary failed in conTeXt when running Luatex, although lualatex worked, as did the i686 binaries on the same machine. In such cases, the easiest solution is to install texlive from source. Similarly, the x86_64 binary version of `asy` runs very slowly when creating 3-D diagrams.

Now, as the `root` user:

```
TEXLIVE_INSTALL_PREFIX=/opt/texlive ./install-tl
```

This command is interactive and allows selection or modification of platform, packages, directories, and other options. The full installation scheme will require about 4.9 gigabytes of disk space. The time to complete the download will depend on your internet connection speed and the number of packages selected.

It has been established by Debian that the `python` scripts in `latex-make` will work with `python3`, so update them to invoke that by running the following command as the `root` user:

```
for F in /opt/texlive/2024/texmf-dist/scripts/latex-make/*.py ; do
    test -f $F && sed -i 's%usr/bin/env python%usr/bin/python3%' $F || true
done
```

Command Explanations

`test -f $F && sed ...`: in a small install these files might not be present, so test if they exist and if not return 'true' to avoid any error if this command has been copied into one of your own install scripts.

`./install-tl --location http://mirror.aut.ac.nz/CTAN/systems/texlive/tlnet/`: use a variation of this if you wish to use a different mirror, for example, because you are in New Zealand but the installer chooses to use an Australian mirror. The list of mirrors is at <https://ctan.org/mirrors/>.

Contents

Installed Programs: Over 300 binaries and symlinks to scripts

Installed Libraries: None

Installed Directories: /opt/texlive

Short Descriptions

TeX programs The programs included in TeX are too numerous to individually list. Please refer to the individual program HTML and PDF pages in the various html, man, or pdf files within the subdirectories of `2024/texmf-dist/`. Using `texdoc pdflatex` (replace `pdflatex` with the command name) may also be useful

texlive-20240312-source

Introduction to TeX Live from source

Note

According to <https://www.tug.org/historic/> the master site in France only supports ftp and rsync. Now that ftp is generally deprecated, that page has links to mirrors, some of which support https, e.g. in Utah and Chemnitz as well as in China. If you prefer to use a different mirror from the example links here, you will need to navigate to systems/historic/texlive/2024 or systems/texlive/2024 as the case may be.

Most of TeX Live can be built from source without a pre-existing installation, but xindy (for indexing) needs working versions of `latex` and `pdflatex` when `configure` is run, and the test suite and install for `asy` (for vector graphics) will fail if TeX has not already been installed. Additionally, biber is not provided within the texlive source and the version of dvisvgm in the texlive tree cannot be built if shared system libraries are used.

All of those packages are dealt with on their own pages and can be built after installing this package. If you have not already done so, you should start at [Setting the PATH for TeX Live](#) so that the final commands to initialize the new installation will be found.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://ftp.math.utah.edu/pub/tex/historic/systems/texlive/2024/texlive-20240312-source.tar.xz>
- Download MD5 sum: 1da2f08e3ba4a3708870dd088c1d6823
- Download size: 67 MB
- Estimated disk space required: 9.1 GB including the additional download and the tests, 8.3 GB installed
- Estimated build time: 4.6 SBU including the additional download and the tests, building with parallelism=4

Required Additional Downloads

Much of the texlive environment (including scripts, documentation, fonts, and various other files) is not part of the source tarball. You must download it separately. This will give you all of the additional files which are provided by a full install of the binary version, as there is no realistic way to restrict which parts get installed.

Because of the size of this package, it is unlikely to be mirrored by BLFS mirrors. If you are unable to download the files for this package, go to <https://www.ctan.org/mirrors/> to find a more-accessible mirror.

- Download (HTTP): <https://ftp.math.utah.edu/pub/tex/historic/systems/texlive/2024/texlive-20240312-texmf.tar.xz>
- Download MD5 sum: e67ce334dd0fddda5f4a87b4fcfaaf48f
- Download size: 4.0 GB

The tlpdb database is shipped as a separate tarball. The `texdoc` program needs a cache file derived from this (and will create the cache on its first run).

- Download (HTTP): <https://ftp.tu-chemnitz.de/pub/tug/historic/systems/texlive/2024/texlive-20240312-extra.tar.xz>
- Download MD5 sum: 1bd045bd1673ce61bde590c10b86c6cc
- Download size: 1.9 MB
- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/texlive-20240312-source-upstream_fixes-1.patch

TeX Live from source Dependencies

Recommended

[a graphical environment](#)

The source ships with its own versions of *many* libraries, and will use them unless it is forced to use the system versions. The following are recommended so that the system version will be used: [Cairo-1.18.0](#), [Fontconfig-2.15.0](#), [FreeType-2.13.3](#), [GC-8.2.6](#), [Graphite2-1.3.14](#), [harfBuzz-9.0.0](#) (built with graphite2 enabled), [ICU-75.1](#), [libpaper-2.2.5](#) (used by at least context and xelatex), and [libpng-1.6.43](#)

Furthermore, the instructions below assume you are using the layout described in [Setting the PATH for TeX Live](#).

Optional

The source ships with its own versions of several libraries which are either not under active development, or only used for limited functionality. If you install these, as with some other optional dependencies in this book you will need to tell `configure` to use the system versions. [GD](#), [t1lib](#), [ZZIPlib](#), [TECkit](#)

Runtime dependencies

[Python2](#) is used by the unmaintained ebong CTAN module (intended for writing Bengali in Rapid Roman Format). `/usr/bin/python` is also in the shebang line for the latex-papersize and lilyglyphs scripts, and documentation at CTAN says both modules have been updated to work with python3. In pythontex there are scripts to invoke python3 or python2 according to the system's version of `python`. [Ruby-3.3.4](#) is used by two scripts, one is for pTeX (Japanese vertical writing) and the other is match_parens which might be generally useful. The perl module [Tk](#), which needs to be run from an X11 session to run the tests and requires [Tk-8.6.14](#) is used by one of the scripts for ptex and is needed for texdoctk (a GUI interface for finding documentation files and opening them with the appropriate viewer). `ps2pdf`, from [ghostscript-10.03.1](#), is used by some utilities and scripts.

Installation of TeX Live

Install TeX Live by running the following commands:

Caution

If you wish to upgrade to current texlive on an older system where extra packages (`asymptote`, `dvisvgm`, or `xindy`) have been installed, you will need to reinstall those as well as fixing up your `PATH` for `$TEXLIVE_PREFIX`.

Note

A successful install requires some texlive commands to be run as the root user, so we will export the `TEXARCH` variable to let `root` use it.

Now, as a normal user:

```
export TEXARCH=$(uname -m | sed -e 's/i.86/i386/' -e 's/$/-linux/') &&
patch -Np1 -i ../texlive-20240312-source-upstream_fixes-1.patch &&
mkdir texlive-build &&
cd texlive-build &&
./configure -C \
--prefix=$TEXLIVE_PREFIX \
--bindir=$TEXLIVE_PREFIX/bin/$TEXARCH \
--datarootdir=$TEXLIVE_PREFIX \
--includedir=$TEXLIVE_PREFIX/include \
--infodir=$TEXLIVE_PREFIX/texmf-dist/doc/info \
--libdir=$TEXLIVE_PREFIX/lib \
--mandir=$TEXLIVE_PREFIX/texmf-dist/doc/man \
--disable-native-texlive-build \
--disable-static --enable-shared \
--disable-dvisvgm \
--with-system-cairo \
--with-system-fontconfig \
--with-system-freetype2 \
--with-system-gmp \
--with-system-graphite2 \
--with-system-harfbuzz \
--with-system-icu \
--with-system-libgs \
--with-system-libpaper \
--with-system-libpng \
--with-system-mpfr \
--with-system-pixman \
--with-system-zlib \
--with-banner-add=" - BLFS" &&
make
```

To test the results, issue: `make -k check`. One test `psutils.test` is known to fail if using system libpaper because TeX Live includes old versions of both psutils and libpaper which result in a difference in the final decimal places of the `psresize` test. The libpaper developer does not think this is significant.

Now, as the `root` user:

```
make install-strip &&
make texlinks &&
```

```
mkdir -pv $TEXLIVE_PREFIX/tlpkg/TeXLive/ &&
install -v -m644 ../../texk/tests/TeXLive/* $TEXLIVE_PREFIX/tlpkg/TeXLive/ &&
tar -xf ../../texlive-20240312-extra.tar.xz -C $TEXLIVE_PREFIX/tlpkg --strip-components=2
```

Note

Only run `make texlinks` once. If it is rerun, it can change all the program symlinks so that they point to themselves and are useless.

Now install the additional files as the `root` user:

```
tar -xf ../../texlive-20240312-texmf.tar.xz -C $TEXLIVE_PREFIX --strip-components=1
```

Still as the `root` user, initialize the new system (the command `fmtutil-sys --all` will produce a *lot* of output):

```
mktextslr &&
fmtutil-sys --all
```

To allow [Evince-46.3.1](#) or [dvisvgm-3.4](#) to link to `libkpathsea.so`, as the `root` user (re)create a symlink from `/usr/lib`:

```
ln -svf $TEXLIVE_PREFIX/lib/libkpathsea.so{,.6} /usr/lib
```

Note

TeX Live does not include the source for ConTeXt, only the items that are at CTAN. Trying to build ConTeXt purely from source on a BLFS TeX Live system is a niche usage and looks as if it will be more trouble than it is worth.

If you wish to use ConTeXt with `luametatable` (most of the old MKII and MKIV code was removed from TeX Live 2023 by the ConTeXt developer), starting with the binary (only installing ConTeXt, 753 MB in March 2024) would probably be the easiest way to bootstrap it. The source for `luametatable` can be pulled with git from [github luametatable](#), compiled with meson and ninja, and installed by copying to the correct program directory. This should avoid having to fix all the symlinks etc and also avoid including the many unrelated parts of `texmf-dist`. See comments 1 to 5 of [#17823](#).

You can now proceed to [asymptote-2.91](#), [biber-2.20](#), [dvisvgm-3.4](#) and / or [xindy-2.5.1](#) if you wish to install them.

Command Explanations

`-c`: creates config.cache, which saves significant time in a parallel build.

`--prefix=`, `--bindir=`, `--datarootdir=`, and other "dir" switches: these switches ensure that the files installed from source will overwrite any corresponding files previously installed by `install-tl-ux` so that the alternative methods of installing `texlive` are consistent.

`--includedir=` `--libdir=`: these switches ensure that the libraries and include files will be within the directories for this year's `texlive`.

`--disable-static`: This switch prevents installation of static versions of the libraries.

`--enable-shared`: Use shared versions of `libkpathsea`, `libptexenc`, `libsynctex`, `libtexlua52` and `libtexluajit`.

`--with-system-...=`: Unless this parameter is used, the included versions of these libraries will be statically compiled into the programs which need them. If you decided not to install a recommended library, omit the corresponding switch.

`--disable-dvisvgm`: As noted above, the shipped version of `dvisvgm`, which has modified configuration files, cannot be built with shared system libraries.

`make texlinks` : this runs the `texlinks.sh` script to create symbolic links from formats to engines. In practice, several of the targets such as `xetex` are now separate binaries and for these it will report "file already exists".

`tar -xf texlive-20240312-texmf.tar.xz -C $TEXLIVE_PREFIX --strip-components=1`: the tarball contains the files for the `texmf-dist` directory, and because of its size we do not want to waste time and space untarring it and then copying the files.

`install -v -m644 ../../texk/tests/TeXLive/* $TEXLIVE_PREFIX/tlpkg/TeXLive/`: This puts the perl modules `TLConfig.pm` and `TLUtils.pm` into the directory where the binary installer puts them - it is at the start of the perl @INC@ PATH within `texlive` when installed using the above configure switches. Without these modules, `texlive` is unusable.

mktexlsr: Create an `ls-R` file which lists what was installed - this is used by kpathsea to find files.

fmtutil-sys --all: This initializes the TeX formats, Metafont bases and Metapost mems.

--without-x: use this if you do not have Xorg installed.

There are many other '--disable' or '--without' options. Some of them such as `--without-texinfo` are accepted but no longer do anything, others will prevent a program being built - but the far greater amount of space used for the related items in texmf means there is no obvious benefit from disabling a few of the programs.

Contents

Installed Programs: Over 300 binaries and symlinks to scripts

Installed Libraries: `libkpathsea.so`, `libptexenc.so`, `libsynctex.so`, `libtexlua52.so`, `libtexluajit.so`

Installed Directories: `$TEXLIVE_PREFIX/bin`, `$TEXLIVE_PREFIX/include`, `$TEXLIVE_PREFIX/lib`, `$TEXLIVE_PREFIX/texmf-dist`

Short Descriptions

<code>TeX programs</code>	The programs included in TeX are too numerous to individually list. Please refer to the individual program HTML and PDF pages in the various html, man, or pdf files within the subdirectories of <code>2024/texmf-dist/</code> . Using <code>texdoc pdflatex</code> (replace <code>pdflatex</code> with the command name) may also be useful
<code>libkpathsea.so</code>	(kpathsearch) exists to look up a file in a list of directories and is used by <code>kpswhich</code>
<code>libptexenc.so</code>	is a library for Japanese pTeX (publishing TeX)
<code>libsynctex.so</code>	is the SyncTeX (Synchronize TeXnology) parser library
<code>libtexlua52.so</code>	provides Lua 5.2, modified for use with LuaTeX
<code>libtexluajit.so</code>	provides LuaJIT, modified for use with LuaJITTeX

asymptote-2.91

Introduction to asymptote

Asymptote is a powerful descriptive vector graphics language that provides a natural coordinate-based framework for technical drawing. Labels and equations can be typeset with LaTeX. As well as EPS, PDF, and PNG output, it can produce WebGL 3D HTML rendering and (using `dvisvgm`) SVG output.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://downloads.sourceforge.net/asymptote/asymptote-2.91.src.tgz>
- Download MD5 sum: `75e09d0dfbf28c26f73401b305427fa7`
- Download size: 14 MB
- Estimated disk space required: 217 MB (46 MB installed after overwriting files in `$TEXLIVE_PREFIX/texmf-dist/`, with all the dependencies which are in BLFS)
- Estimated build time: 0.9 SBU (using parallelism=4)

asymptote Dependencies

Required

[ghostscript-10.03.1](#) and [texlive-20240312](#)

Recommended

[cURL-8.9.1](#), [Freeglut-3.6.0](#), [GC-8.2.6](#), [GLEW-2.2.0](#), [GLM-1.0.1](#) and [libtirpc-1.3.5](#)

Recommended at runtime

[dvisvgm-3.4](#) to allow svg output

Optional

[fftw-3.3.10](#), [gsl-2.8](#), [libsigsegv-2.14](#), both [Boost-1.86.0](#) and [rapidjson](#) to use [LSP](#) which can be used with [emacs](#), and [eigen](#)

Optional (at runtime)

[ImageMagick-7.1.1-36](#) to convert output to other formats such as JPEG or to create animated GIFs, [PyQt5](#) (not tested, has a build dependency of [qt5-components-5.15.14](#)) to use xasy.

Editor Notes: <https://wiki.linuxfromscratch.org/blfs/wiki/asymptote>

Installation of asymptote

Note

Certain 3D PDF scripts may not work when invoked from pdflatex on some zen+ amdgpu APU machines.

Install asymptote by running the following commands:

```
export TEXARCH=$(uname -m | sed -e 's/i.86/i386/' -e 's/$/-linux/') &&  
  
.configure --prefix=$TEXLIVE_PREFIX  
          --bindir=$TEXLIVE_PREFIX/bin/$TEXARCH  
          --datarootdir=$TEXLIVE_PREFIX/texmf-dist  
          --infodir=$TEXLIVE_PREFIX/texmf-dist/doc/info  
          --libdir=$TEXLIVE_PREFIX/texmf-dist  
          --mandir=$TEXLIVE_PREFIX/texmf-dist/doc/man  
          --disable-lsp  
          --enable-gc=system  
          --with-latex=$TEXLIVE_PREFIX/texmf-dist/tex/latex \  
          --with-context=$TEXLIVE_PREFIX/texmf-dist/tex/context/third &&  
  
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

`--prefix=`, `--bindir=`, `--datarootdir=`, and other "dir" switches: these switches ensure that the files installed from source will overwrite any corresponding files previously installed by install-tl-unx so that the alternative methods of installing texlive are consistent.

`--libdir=$TEXLIVE_PREFIX/texmf-dist`: This parameter ensures that the `asymptote` directory will similarly overwrite any files installed by install-tl-unx.

`--disable-lsp`: The Language Server Protocol is enabled by default, with several third-party sources included, but it does not build unless boost and rapidjson have been installed.

`--enable-gc=system`: this ensures that the system version of `libgc.so` will be used instead of the version shipped with this package.

`--with-latex=` `--with-context=`: These switches ensure that style files and a tex file will be installed into the texlive directories instead of creating a `texmf-local` directory for them.

Contents

Installed Programs: asy and xasy

Installed Libraries: None

Installed Directory: `$TEXLIVE_PREFIX/texmf-dist/asymptote`, `$TEXLIVE_PREFIX/doc/asymptote`, and `$TEXLIVE_PREFIX/info/asymptote`

Short Descriptions

`asy` is a vector graphics program

`xasy` is a Python3 script providing a Qt5 GUI for asy

biber-2.20

Introduction to biber

Biber is a BibTeX replacement for users of biblatex, written in Perl, with full Unicode support.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/plk/biber/archive/v2.20/biber-2.20.tar.gz>
- Download MD5 sum: a7f400cdac3a3d9d9ab13ca053eb6791
- Download size: 1.6 MB
- Estimated disk space required: 15 MB (installs 1.6MB of perl modules)
- Estimated build time: 0.4 SBU including tests

Note

If you have updated to a new release of texlive-source, biber and its perl dependencies are still present in `/usr` and do not need to be rebuilt. However, biblatex installs in `/opt/texlive/2024` and does need to be reinstalled.

If you installed the above dependencies using the BLFS instructions for each of them, `perl ./Build.PL` will complain that Mozilla::CA is not installed, but that is not needed unless CPAN was used to install the modules. BLFS patches LWP::Protocol::https to use the system certificates, Mozilla::CA uses old certificates.

Biber Dependencies

Required

[autovivification-0.18](#), [Business-ISBN-3.009](#), [Business-ISMN-1.204](#), [Business-ISSN-1.005](#), [Class-Accessor-0.51](#), [Data-Compare-1.29](#), [Data-Dump-1.25](#), [Data-Uniqid-0.12](#), [DateTime-Calendar-Julian-0.107](#), [DateTime-Format-Builder-0.83](#), [Encode-EUCJPASCII-0.03](#), [Encode-HanExtra-0.23](#), [Encode-JIS2K-0.05](#), [File-Slurper-0.014](#), [IO-String-1.08](#), [IPC-Run3-0.049](#), [Lingua-Translit-0.29](#), [List-AllUtils-0.19](#), [List-MoreUtils-0.430](#), [Log-Log4perl-1.57](#), [LWP-Protocol-https-6.14](#), [Module-Build-0.4234](#), [Parse-RecDescent-1.967015](#), [PerlIO-utf8 strict-0.010](#), [Regexp-Common-2024080801](#), [Sort-Key-1.33](#), [Text-BibTeX-0.89](#), [Text-CSV-2.04](#), [Text-Roman-3.5](#), [Unicode-Collate-1.31](#), [Unicode-LineBreak-2019.001](#), [XML-LibXML-Simple-1.01](#), [XML-LibXSLT-2.003000](#), and [XML-Writer-0.900](#)

Recommended (required for the test suite)

[File-Which-1.27](#) and [Test-Differences-0.71](#)

Note

It is possible to install all missing dependencies automatically. You must first install [Module-Build-0.4234](#) using [automatic installation of perl modules](#). Then run `perl ./Build.PL` and when it prompts you, become the `root` user and run `./Build installdeps` - this will use CPAN and as noted above it will use Mozilla::CA instead of using system certificates.

Installation of Biber

Install Biber by running the following commands:

```
perl ./Build.PL &&
./Build
```

To test the results, enter: `./Build test`

Now, as the `root` user:

```
./Build install
```

Contents

Installed Programs: biber

Installed Library: None

Installed Directory: /usr/lib/perl5/site_perl/5.*{,<arch>}-linux/auto}/Biber

Short Descriptions

biber is used for producing bibliographies in LaTeX documents

dvisvgm-3.4

Introduction to dvisvgm

The dvisvgm package converts DVI, EPS and PDF files to SVG format.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://github.com/mgrieseki/dvisvgm/releases/download/3.4/dvisvgm-3.4.tar.gz>
- Download MD5 sum: 8d881999c56c5811b35ebaf86206a1e1
- Download size: 2.8 MB
- Estimated disk space required: 268 MB (net 7 MB installed after overwriting files in \$TEXLIVE_PREFIX/texmf-dist/, add 948 MB for the tests)
- Estimated build time: 0.8 SBU (add 0.5 SBU for the tests, both using parallelism=4)

dvisvgm Dependencies

Required

[Brotli-1.1.0](#), [ghostscript-10.03.1](#), [Potrace-1.16](#), and [texlive-20240312](#)

Recommended

[WOFF2-1.0.2](#)

Optional

[asciidoc-10.2.1](#) with [xmlto-0.0.29](#) and [libxslt-1.1.42](#), [dblatex](#) (with the above) and [xxHash](#) (the current version is included in the tarball)

Installation of dvisvgm

The test suite assumes that a modern version of Python has been installed as plain `python`. Changing this to use `python3` requires a sed to one `Makefile.in` file.

Install dvisvgm by running the following commands:

```
sed -i 's/python/&3/' tests/Makefile.in      &&
./configure
  --bindir=$TEXLIVE_PREFIX/bin/${TEXARCH} \
  --mandir=$TEXLIVE_PREFIX/texmf-dist/doc/man \
  --with-kpathsea=$TEXLIVE_PREFIX            &&
make
```

To test the results, issue: `make check`.

Now, as the `root` user:

```
make install
```

Command Explanations

--with-kpathsea=\$TEXLIVE_PREFIX: This allows the build system to find the headers for kpathsea

Contents

Installed Programs: dvisvgm

Installed Libraries: None

Installed Directories: None

Short Descriptions

dvisvgm converts DVI, EPS and PDF files to the SVG format

xindy-2.5.1

Introduction to xindy

Xindy is an index processor that can be used to generate book-like indexes for arbitrary document-preparation systems. This includes systems such as TeX and LaTeX, the roff-family, and SGML/XML-based systems (e.g., HTML) that process some kind of text and generate indexing information.

This package is known to build and work properly using an LFS 12.2 platform.

Package Information

- Download (HTTP): <https://tug.ctan.org/support/xindy/base/xindy-2.5.1.tar.gz>
- Download MD5 sum: 221acf6b0f6f8388f89a59c56491041
- Download size: 506 KB
- Estimated disk space required: 15 MB
- Estimated build time: less than 0.1 SBU

Required Additional Downloads

- Required patch: https://www.linuxfromscratch.org/patches/blfs/12.2/xindy-2.5.1-upstream_fixes-2.patch

Xindy Dependencies

Required

[Clisp-2.49](#) and [texlive-20240312](#)

Installation of xindy

Install xindy by running the following commands:

```
export TEXARCH=$(uname -m | sed -e 's/i.86/i386/' -e 's/$/-linux/') &&
sed -i "s/ grep -v '^;/'/ awk NF/" make-rules/inputenc/Makefile.in &&
sed -i 's%\(\indexentry\)%\1\\%' make-rules/inputenc/make-inp-rules.pl &&
patch -Np1 -i ../../xindy-2.5.1-upstream_fixes-2.patch &&
./configure --prefix=$TEXLIVE_PREFIX \
            --bindir=$TEXLIVE_PREFIX/bin/$TEXARCH \
            --datarootdir=$TEXLIVE_PREFIX \
            --includedir=/usr/include \
            --libdir=$TEXLIVE_PREFIX/texmf-dist \
            --mandir=$TEXLIVE_PREFIX/texmf-dist/doc/man &&
make LC_ALL=POSIX
```

This package does not have a test suite.

Now, as the `root` user:

`make install`

Command Explanations

`sed -i "s/ grep -v '^';' / awk NF/" ...`: The build sorts files in latin{1..3} encodings to create latin.xdy, and unicode versions of these to create utf8.xdy after using `grep -v '^';'` to remove blank lines. With grep-2.23 any data not in the expected encoding is treated as binary, resulting in a useless file. This command uses an alternative way of removing blank lines.

`sed -i 's%\\(indexentry\\)%\\\\%'` ...: A regexp contains `indexentry{` - perl has warned about the unescaped left brace for some time and now treats it as illegal. Change it to `indexentry\\{`, doubling the backslash for `sed`.

`patch -Np1 -i ../xindy-2.5.1-upstream_fixes-2.patch`: Xindy is now maintained at CTAN. This patch updates the source with some of the changes made there (but ignoring changes which were only made to allow for spaces in pathnames and some trivial recent changes).

`--prefix=`, `--bindir=`, `--datarootdir=`, and other "dir" switches: these switches ensure that the files installed from source will overwrite any corresponding files previously installed by `install-tl-unx` so that the alternative methods of installing texlive are consistent.

`--includedir=/usr/include`: This parameter ensures that the `kpathsea` headers from [texlive-20240312](#) will be found.

`make LC_ALL=POSIX`: with the current version of coreutils it is essential to build xindy in the POSIX (or C) locale because in a UTF-8 locale the file `latin.xdy` will contain only a heading and then a line '`Binary file (standard input) matches`' instead of the many lines of lisp merge-rule commands it ought to contain.

Contents

Installed Programs: `tex2xindy`, `texindy`, `xindy`

Installed Libraries: None

Installed Directory: `$TEXLIVE_PREFIX/texmf-dist/xindy`

Short Descriptions

<code>tex2xindy</code>	transforms a LaTeX index file into a xindy raw index file
<code>texindy</code>	is a wrapper for xindy that turns on many LaTeX conventions by default
<code>xindy</code>	creates a sorted and tagged index from a raw LaTeX index

Appendix A. Creative Commons License

Creative Commons Legal Code

Attribution-NonCommercial-ShareAlike 2.0

Important

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

- a. "Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

- b. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
 - c. "Licensor" means the individual or entity that offers the Work under the terms of this License.
 - d. "Original Author" means the individual or entity who created the Work.
 - e. "Work" means the copyrightable work of authorship offered under the terms of this License.
 - f. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
 - g. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, Noncommercial, ShareAlike.
2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.
3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
- a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
 - b. to create and reproduce Derivative Works;
 - c. to distribute copies or phonorecords of, display publicly, perform publicly, and conduct publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
 - d. to distribute copies or phonorecords of, display publicly, perform publicly, and conduct publicly by means of a digital audio transmission Derivative Works;

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights set forth in Sections 4(e) and 4(f).

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:
- a. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any reference to such Licensor or the Original Author, as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any reference to such Licensor or the Original Author, as requested.
 - b. You may distribute, publicly display, publicly perform, or publicly digitally perform a Derivative Work only under the terms of this License, a later version of this License with the same License Elements as this License, or a Creative Commons iCommons license that contains the same License Elements as this License (e.g. Attribution-NonCommercial-ShareAlike 2.0 Japan). You must include a copy of, or the Uniform Resource Identifier for, this License or other license specified in the previous sentence with every copy or phonorecord of each Derivative Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Derivative Works that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder, and You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Derivative Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Derivative Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Derivative Work itself to be made subject to the terms of this License.
 - c. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or

directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.

- d. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and give the Original Author credit reasonable to the medium or means You are utilizing by conveying the name (or pseudonym if applicable) of the Original Author if supplied; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.
- e. For the avoidance of doubt, where the Work is a musical composition:
 - i. Performance Royalties Under Blanket Licenses. Licensor reserves the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work if that performance is primarily intended for or directed toward commercial advantage or private monetary compensation.
 - ii. Mechanical Rights and Statutory Royalties. Licensor reserves the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions), if Your distribution of such cover version is primarily intended for or directed toward commercial advantage or private monetary compensation.
 - iii. 6. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor reserves the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions), if Your public digital performance is primarily intended for or directed toward commercial advantage or private monetary compensation.
- f. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor reserves the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions), if Your public digital performance is primarily intended for or directed toward commercial advantage or private monetary compensation.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
- b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

- a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
- b. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
- c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this

- agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
- d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
 - e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Important

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, neither party will use the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time.

Creative Commons may be contacted at <http://creativecommons.org/>.

Appendix B. The MIT License

Copyright © 1999-2024 The BLFS Development Team

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Glossary

Acronyms

669

UNIS/Composer 669 Module

ABI

Application Binary Interface

ADSL

Asymmetric Digital Subscriber Line

AFS

Andrew File System

AGESA

AMD Generic Encapsulated Software Architecture

AIFF

Audio Interchange File Format

ALSA

Advanced Linux Sound Architecture

ANSI

American National Standards Institute

API

Application Programming Interface

APR

Apache Portable Runtime

ARP

Address Resolution Protocol

ASCII

American Standard Code for Information Interchange

ASLR

Address Space Layout Randomization

ASN

Abstract Syntax Notation

ASF

Advanced Streaming Format

ATA

Advanced Technology Attachment

ATSC

Advanced Television Systems Committee

ATK

Accessibility ToolKit

AVI

Audio Video Interleave

AWT

Abstract Window Toolkit

BER

Basic Encoding Rules

BIND

Berkeley Internet Name Domain

BIOS

Basic Input/Output System

BLFS

Beyond Linux From Scratch

BMP

Bit MaP

CD

Compact Disk

CDDA

Compact Disc Digital Audio

CIFS

Common Internet File System

See Also [SMB](#).

CMS

Cryptographic Message Syntax

CODEC

COmpression/DECompression module

CORBA

Common Object Request Broker Architecture

CPU

Central Processing Unit

CRD

Color Rendering Dictionary

CSA

Color Space Array

CSS (on DVD)

Contents Scrambling System

CSS

Cascading Style Sheets

CUPS

Common Unix Printing System

CVS

Concurrent Versions System

DAO

Disc At Once

DARPA

Directory Address Resolution Protocol Allocation

DEC

Digital Equipment Corporation

DER

Distinguished Encoding Rules

DES

Data Encryption Standard

DHCP

Dynamic Host Configuration Protocol

DICT

Dictionary Server Protocol (RFC 2229)

DIN

German Industrial Norm

DNS

Domain Name Service

DOS

Disk Operating System

DRI

Direct Rendering Infrastructure

DSC

Document Structuring Conventions

DSO

Dynamic Shared Objects

DSSSL

Document Style Semantics and Specification Language

DV

Digital Video

DVD

Digital Versatile Disk (also Digital Video Disk)

DVI

DeVice Independent

ELF

Executable and Linking Format

EPP

Enhanced Parallel Port

EPS

Encapsulated PostScript

ESD

Enlighten Sound Daemon

ESMTP

Extended Simple Mail Transfer Protocol

FAM

File Alteration Monitor

FAME

Fast Assembly Mpeg Encoder

FAQ

Frequently Asked Questions

FAX

Facsimile

FB

Frame Buffer

FHS

File Hierarchy Standard

FLAC

Free Lossless Audio CODEC

FO

Formatted Objects

FTP

File Transfer Protocol

GCC

GNU Compiler Collection

GDBM
GNU DataBase Manager

GDK
GTK+ Drawing Kit

GDM
GNOME Display Manager

GID
Group IDentity

GIF
Graphics Interchange Format

GLUT
OpenGL Utility Toolkit

GMP
GNU Multiple Precision Arithmetic

GNAT
GNU NYU Ada 9x Translator

GNOME
GNU Network Object Model Environment

GNU
GNU's Not Unix

GOT
Global Offset Table

GPL
General Public License

GPM
General Purpose Mouse

GSS
Generic Security Service

GSSAPI
Generic Security Service Application Programming Interface

GTK
GIMP ToolKit

GUI
Graphical User Interface

HFS
Hierarchical File System

HTML
HyperText Markup Language

HTTP
HyperText Transfer Protocol

HTTPS
HyperText Transfer Protocol Secured

HUP
Hang UP

IANA
Internet Assigned Numbers Authority

ICC
International Color Consortium

ICMP
Internet Control Message Protocol

IDE
Integrated Drive Electronics
Integrated Development Environment

IDL
Interface Definition Language

IJS
Ink Jet Systems

ILS
Internet Location Server

IMAP
Internet Message Access Protocol

IMON
Inode MONitor

IP
Internet Protocol
See Also [TCP](#) .

IPX
Internetwork Packet eXchange

IRC
Internet Relay Chat

ISDN
Integrated Services Digital Network

ISO
International Standards Organisation

ISP
Internet Service Provider

IT
ImpulseTracker Module

JAI
Java Advanced Imaging

JAR
Java ARchive

JDK
Java Development Kit

JFIF
JPEG File Interchange Format

JPEG
Joint Photographic Experts Group

KDC
Key Distribution Center

KDE
KDesktop Environment

LAME
Lame Ain't an MP3 Encoder

LAN
Local Area Network

LDAP
Lightweight Directory Access Protocol

LDIF
Lightweight Data Interchange Format

LFS
Linux From Scratch

LGPL
Library General Public License

LPR
Line PRinter

LZO
Lempel-Ziv-Oberhumer

LZW
Lempel-Ziv-Welch

MAC
Media Access Control

MCOP
Multimedia COmmunication Protocol

MCU
Multipoint Control Unit

MD
Message-Digest

MDA
Mail Delivery Agent

MED
MED/OctaMED Module

MIDI

Musical Instrument Digital Interface

MIF

Marker Interchange Format

MII

Media Independent Interface

MIME

Multipurpose Internet Mail Extensions

MIT

Massachusetts Institute of Technology

MNG

Multiple-image Network Graphics

MOD

ProTracker Module

MP3

MPEG-1 audio layer 3

MPEG

Moving Picture Experts Group

MSL

Magick Scripting Language

MTA

Mail Transport Agent

MTM

MultiTracker Module

MUA

Mail User Agent

NASM

Netwide ASseMbler

NNTP

Network News Transfer Protocol

NFS

Network File System

NIS

Network Information Service

NPTL

Native Posix Thread Library

NSPR

Netscape Portable Runtime

NSS

Network Security Services

NTP

Network Time Protocol

OAF

Object Activation Framework

ODBC

Open DataBase Connectivity

OMF

Open Metadata Framework

ORB

Object Request Broker

See Also [CORBA](#).

ORDBMS

Object Relational Database Management System

OS

Operating System

OSF

Open Software Foundation

OSS

Open Sound System

PAM

Pluggable authentication Modules

PBM
Portable BitMap

PCI
Peripheral Component Interconnect

PCL
Printer Control Language

PCM
Pulse Code Modulation

PDC
Primary Domain Controller

PDF
Portable Document Format

PEAR
PHP Extension and Application Repository

PGM
Portable Grey Map

PGP
Pretty Good Privacy

PHP
PHP Hypertext Preprocessor

PIE
Position Independent Executable

PIM
Personal Information Manager

PLIP
Parallel Line Internet Protocol

PNG
Portable Network Graphics

PO
Portable Object

POD
Plain Old Documentation

POP
Post Office Protocol

PPD
PostScript Printer Description

PPM
Portable Pixel Map

PPP
Point-to-Point Protocol

PPPoE
Point-to-Point Protocol over Ethernet

PS
PostScript

RADIUS
Remote Authentication Dial-In User Service

RAM
Random Access Memory

RARP
Reverse Address Resolution Protocol

RCS
Revision Control System

RFC
Request For Comments

RGB
Red Green Blue

RGBA
Red Green Blue Alpha

ROM
Read-Only Memory

RP

Roaring Penguin

RPC

Remote Procedure Call

RTC

Real Time Clock

RTP

Real Time Protocol

RW

Read Write

S3M

ScreamTracker Version 3 Module

S/MIME

Secure/MIME

SANE

Scanner Access Now Easy

SASL

Simple Authentication and Security Layer

SATA

Serial Advanced Technology Attachment

SBU

Standard Build Unit

SCSI

Small Computer System Interface

SDK

Software Development Kit

SGML

Standard Generalized Markup Language

SMART

Self Monitoring Analysis and Reporting Technology

SMB

Server Message Block

SMIL

Synchronized Multimedia Integration Language

SMTP

Simple Mail Transfer Protocol

SQL

Structured Query Language

SSH

Secure SHell

SSL

Secure Sockets Layer

SSP

Stack Smash Protection

SUID

Set User IDentity

SVG

Scalable Vector Graphics

SVGA

Super Video Graphics Array

TCL

Tool Command Language

TCP

Transmission Control Protocol

TGT

Ticket-Granting Ticket

TIFF

Tag(ed) Image File Format

TLS

Transport Layer Security

TTF

TrueType Font

TTS
Text To Speech

UCS
Universal Character Set

UDF
Universal Disk Format

UID
User IDentity

UDP
User Datagram Protocol

UI
User Interface

UML
Unified Modelling Language

URL
Uniform Resource Locator

USB
Universal Serial Bus

USR
Upstream Ready

UTF
UCS Transformation Format

UUCP
Unix-to-Unix Copy Protocol

VCD
Video Compact Disk

VCS
Version Control System

VESA
Video Electronics Standards Association

VGA
Video Graphics Array

VNC
Virtual Network Computer

VOB
Video OBject

VOIP
Voice Over IP

W3C
World Wide Web Consortium

WAV
Waveform Audio

WWW
World Wide Web

XDMCP
XDisplay Manager Control Protocol

XM
FastTracker Module

XML
eXtensible Markup Language

XSL
eXtensible Style Language

XSLT
eXtensible Style Language Transformation

XSM
X/Open System Management

XMMS
XMuliMedia System

YP
Yellow Pages

YUV

Index

Packages

AAlib: [AAlib-1.4rc5](#)
AbiWord: [AbiWord-3.0.5](#)
Abseil-cpp: [Abseil-cpp-20240722.0](#)
AccountsService: [AccountsService-23.13.9](#)
acpid: [acpid-2.0.34](#)
adwaita-icon-theme: [adwaita-icon-theme-46.2](#)
alabaster: [Alabaster-1.0.0](#)
Algorithm::Diff: [Algorithm::Diff-1.201](#)
Alien::Build: [Alien::Build-2.83](#)
Alien::Build::Plugin::Download::GitLab: [Alien::Build::Plugin::Download::GitLab-0.01](#)
Alien::Libxml2: [Alien::Libxml2-0.19](#)
alsa-firmware: [alsa-firmware-1.2.4](#)
alsa-lib: [alsa-lib-1.2.12](#)
alsa-plugins: [alsa-plugins-1.2.12](#)
alsa-tools: [alsa-tools-1.2.11](#)
alsa-utils: [alsa-utils-1.2.12](#)
Apache: [Apache-2.4.62](#)
Apache Ant: [apache-ant-1.10.14](#)
AppStream: [AppStream-1.0.3](#)
appstream-glib: [appstream-glib-0.8.3](#)
Apr: [Apr-1.7.4](#)
Apr-Util: [Apr-Util-1.6.3](#)
Archive::Zip: [Archive::Zip-1.68](#)
Ark: [ark-24.08.0](#)
asciidoc: [Asciidoc-10.2.1](#)
asciidocctor: [Asciidocctor-2.0.23](#)
Aspell: [Aspell-0.60.8.1](#)
asymptote: [asymptote-2.91](#)
at: [at-3.2.5](#)
at-spi2-core: [at-spi2-core-2.52.0](#)
Atkmm: [Atkmm-2.28.4](#)
Atkmm: [Atkmm-2.36.3](#)
attr: [Attrs-24.2.0](#)
Audacious: [Audacious-4.4](#)
AudioFile: [AudioFile-0.3.6](#)
Autofs: [autofs-5.1.9](#)
autovivification: [autovivification-0.18](#)
Avahi: [Avahi-0.8](#)
B::COW: [B::COW-0.007](#)
B::Hooks::EndOfScope: [B::Hooks::EndOfScope-0.28](#)
Babel: [babel-2.16.0](#)
Babl: [babl-0.1.108](#)
Balsa: [Balsa-2.6.4](#)
Baobab: [Baobab-46.0](#)
Biber: [biber-2.20](#)
BIND: [BIND-9.20.0](#)
BIND Utilities: [BIND Utilities-9.20.0](#)
BLFS Systemd Units: [BLFS Systemd Units](#)
Bluefish: [Bluefish-2.2.15](#)
BlueZ: [BlueZ-5.77](#)
Bogofilter: [Bogofilter-1.2.5](#)
Boost: [boost-1.86.0](#)
Brasero: [Brasero-3.12.3](#)
breeze-icons: [breeze-icons-6.5.0](#)
bridge-utils: [bridge-utils-1.7.1](#)
Brotli: [brotli-1.1.0](#)
btrfs-progs: [btrfs-progs-6.10.1](#)
bubblewrap: [Bubblewrap-0.9.0](#)
Business::ISBN: [Business::ISBN-3.009](#)
Business-ISBN-Data: [Business::ISBN::Data-20240807.001](#)
Business::ISMN: [Business::ISMN-1.204](#)
Business::ISSN: [Business::ISSN-1.005](#)
c-ares: [c-ares-1.33.0](#)
cachecontrol: [CacheControl-0.14.0](#)
Cairo: [Cairo-1.18.0](#)
Cantarell fonts: [TTF and OTF fonts -- description](#)
Capture::Tiny: [Capture::Tiny-0.48](#)

cbindgen: [Cbindgen-0.27.0](#)
CDParanoia: [CDParanoia-III-10.2](#)
Cdrdao: [Cdrdao-1.2.4](#)
Cdrtools: [Cdrtools-3.02a09](#)
certifi: [certifi-2024.7.4](#)
chardet: [Chardet-5.2.0](#)
charset-normalizer: [Charset-normalizer-3.3.2](#)
cifs-utils: [cifs-utils-7.0](#)
Class::Accessor: [Class::Accessor-0.51](#)
Class::Data::Inheritable: [Class::Data::Inheritable-0.09](#)
Class::Inspector: [Class::Inspector-1.36](#)
Class::Singleton: [Class::Singleton-1.6](#)
Class::Tiny: [Class::Tiny-1.008](#)
Clisp: [Clisp-2.49](#)
Clone: [Clone-0.46](#)
clucene: [CLucene-2.3.3.4](#)
CMake: [CMake-3.30.2](#)
Colord: [Colord-1.4.7](#)
colord-gtk: [colord-gtk-0.3.1](#)
commonmark: [Commonmark-0.9.1](#)
Compface: [Compface-1.5.2](#)
Config::AutoConf: [Config::AutoConf-0.320](#)
console-fonts: [About Console Fonts](#)
CPAN automated install of perl modules: [CPAN automated install of perl modules](#)
CPAN::Meta::Check: [CPAN::Meta::Check-0.018](#)
Cpio: [cpio-2.15](#)
CrackLib: [CrackLib-2.10.2](#)
cryptsetup: [cryptsetup-2.7.4](#)
cssselect: [CSSSelect-1.2.0](#)
Cups: [Cups-2.4.10](#)
cups-browsed: [cups-browsed-2.0.1](#)
cups-filters: [cups-filters-2.0.1](#)
cups-pk-helper: [cups-pk-helper-0.2.7](#)
cURL: [cURL-8.9.1](#)
Cyrus SASL: [Cyrus SASL-2.1.28](#)
cython: [Cython-3.0.11](#)
D-Bus: [dbus-1.14.10](#)
Dash: [Dash-0.5.12](#)
Data::Compare: [Data::Compare-1.29](#)
Data::Dump: [Data::Dump-1.25](#)
Data::Uniqid: [Data::Uniqid-0.12](#)
DateTime: [DateTime-1.65](#)
DateTime::Calendar::Julian: [DateTime::Calendar::Julian-0.107](#)
DateTime::Format::Builder: [DateTime::Format::Builder-0.83](#)
DateTime::Format::Strptime: [DateTime::Format::Strptime-1.79](#)
DateTime::Locale: [DateTime::Locale-1.43](#)
DateTime::TimeZone: [DateTime::TimeZone-2.62](#)
dbus-glib: [dbus-glib-0.112](#)
dbus-python: [D-Bus Python-1.3.2](#)
dbusmock: [dbusmock-0.32.1](#)
DConf: [DConf-0.40.0 / DConf-Editor-45.0.1](#)
DejaVu fonts: [TTF and OTF fonts -- description](#)
desktop-file-utils: [desktop-file-utils-0.27](#)
Devel::StackTrace: [Devel::StackTrace-2.05](#)
dhpcd: [dhpcd-10.0.8](#)
Dist::CheckConflicts: [Dist::CheckConflicts-0.11](#)
DocBook DSSSL Stylesheets: [docbook-dsssl-1.79](#)
DocBook SGML DTD-3.1: [docbook-3.1-dtd](#)
DocBook SGML DTD-4.5: [docbook-4.5-dtd](#)
DocBook-utils: [DocBook-utils-0.6.14](#)
DocBook XML DTD: [docbook-xml-4.5](#)
DocBook XSL Stylesheets: [docbook-xsl-nons-1.79.2](#)
DocBook XML DTD and Schemas 5.0: [docbook-xml-5.0](#)
DocBook XML Schemas 5.1: [docbook-xml-5.1](#)
docutils: [docutils-0.21.2](#)
dolphin: [dolphin-24.08.0](#)
dolphin-plugins: [dolphin-plugins-24.08.0](#)
dos2unix: [dos2unix-7.5.2](#)
dosfstools: [dosfstools-4.2](#)
double-conversion: [Double-conversion-3.3.0](#)
Dovecot: [Dovecot-2.3.21.1](#)
Doxygen: [Doxygen-1.12.0](#)
doxypypp: [Doxypypp-0.8.8.7](#)
doxyqml: [Doxyqml-0.5.3](#)
dtc: [dtc-1.7.1](#)
duktape: [duktape-2.7.0](#)

dvd+rw-tools: [dvd+rw-tools-7.1](#)
dvisvgm: [dvisvgm-3.4](#)
Ed: [Ed-1.20.2](#)
editables: [Editables-0.5](#)
efibootmgr: [efibootmgr-18](#)
efivar: [efivar-39](#)
Emacs: [Emacs-29.4](#)
enchant: [enchant-2.8.2](#)
Encode::EUCJPASCII: [Encode::EUCJPASCII-0.03](#)
Encode::HanExtra: [Encode::HanExtra-0.23](#)
Encode::JIS2K: [Encode::JIS2K-0.05](#)
Encode::Locale: [Encode::Locale-1.05](#)
Enscript: [Enscript-1.6.6](#)
EOG: [EOG-45.4](#)
epdfview: [ePDFView-gtk3-20200814](#)
Epiphany: [Epiphany-46.3](#)
Eval::Closure: [Eval::Closure-0.14](#)
Evince: [Evince-46.3.1](#)
Evolution: [Evolution-3.52.4](#)
evolution-data-server: [evolution-data-server-3.52.4](#)
Exception::Class: [Exception::Class-1.45](#)
Exempi: [Exempi-2.6.5](#)
Exim: [Exim-4.98](#)
Exiv2: [Exiv2-0.28.3](#)
Exo: [Exo-4.18.0](#)
Exporter::Tiny: [Exporter::Tiny-1.006002](#)
extra-cmake-modules: [extra-cmake-modules-6.5.0](#)
ExtUtils::LibBuilder: [ExtUtils::LibBuilder-0.08](#)
faac: [FAAC-1_30](#)
faad2: [FAAD2-2.11.1](#)
falkon: [falkon-24.08.0](#)
Fcron: [Fcron-3.2.1](#)
fdk-aac: [fdk-aac-2.0.3](#)
feh: [feh-3.10.3](#)
Fetchmail: [Fetchmail-6.4.39](#)
FFI::CheckLib: [FFI::CheckLib-0.31](#)
FFmpeg: [FFmpeg-7.0.2](#)
fftw: [fftw-3.3.10](#)
File::chdir: [File::chdir-0.1011](#)
File::Copy::Recursive: [File::Copy::Recursive-0.45](#)
File::FcntlLock: [File::FcntlLock-0.22](#)
File::Find::Rule: [File::Find::Rule-0.34](#)
File::Listing: [File::Listing-6.16](#)
File::Roller: [File::Roller-44.3](#)
File::ShareDir: [File::ShareDir-1.118](#)
File::ShareDir::Install: [File::ShareDir::Install-0.14](#)
File::Slurper: [File::Slurper-0.014](#)
File::Which: [File::Which-1.27](#)
Firefox: [Firefox-128.1.0esr](#)
FLAC: [FLAC-1.4.3](#)
FLTK: [FLTK-1.3.9](#)
Fluxbox: [Fluxbox-1.3.7](#)
fmt: [fmt-11.0.2](#)
Fontconfig: [Fontconfig-2.15.0](#)
FontForge: [FontForge-20230101](#)
fop: [fop-2.9](#)
Freeglut: [Freeglut-3.6.0](#)
FreeRDP: [FreeRDP-3.7.0](#)
FreeType: [FreeType-2.13.3](#)
frei0r: [frei0r-plugins-1.8.0](#)
FriBidi: [FriBidi-1.0.15](#)
Fuse 3: [Fuse-3.16.2](#)
Garcon: [Garcon-4.18.2](#)
Gavl: [gavl-1.4.0](#)
GC: [GC-8.2.6](#)
GCC-14.2.0: [GCC-14.2.0](#)
Gcr: [Gcr-3.41.2](#)
Gcr: [Gcr-4.3.0](#)
GDB: [GDB-15.1](#)
gdk-pixbuf: [gdk-pixbuf-2.42.12](#)
GDM: [GDM-46.2](#)
Gedit: [Gedit-47.0](#)
Gegl: [gegl-0.4.48](#)
GeoClue: [GeoClue-2.7.1](#)
geocode-glib: [geocode-glib-3.26.4](#)
gexiv2: [gexiv2-0.14.3](#)

Ghostscript: [ghostscript-10.03.1](#)
gi-docgen: [Gi-DocGen-2024.1](#)
giflib: [giflib-5.2.2](#)
Gimp: [Gimp-20240711](#)
Git: [Git-2.46.0](#)
Gjs: [Gjs-1.80.2](#)
Glad: [Glad-2.0.6](#)
GLEW: [GLEW-2.2.0](#)
glib-networking: [glib-networking-2.80.0](#)
GLib2: [GLib2-2.80.4](#)
GLibmm: [GLibmm-2.66.7](#)
GLibmm: [GLibmm-2.80.0](#)
GLM: [GLM-1.0.1](#)
Glslang: [Glslang-14.3.0](#)
GLU: [GLU-9.0.3](#)
GMime 3: [GMime-3.2.7](#)
gnome-autoar: [gnome-autoar-0.4.4](#)
gnome-backgrounds: [gnome-backgrounds-46.0](#)
gnome-bluetooth: [gnome-bluetooth-46.1](#)
gnome-calculator: [gnome-calculator-46.1](#)
gnome-color-manager: [gnome-color-manager-3.36.0](#)
gnome-connections: [gnome-connections-46.0](#)
gnome-control-center: [gnome-control-center-46.4](#)
gnome-desktop: [gnome-desktop-44.1](#)
gnome-disk-utility: [gnome-disk-utility-46.0](#)
gnome-icon-theme: [gnome-icon-theme-3.12.0](#)
gnome-icon-theme-extras: [gnome-icon-theme-extras-3.12.0](#)
gnome-icon-theme-symbolic: [gnome-icon-theme-symbolic-3.12.0](#)
gnome-keyring: [gnome-keyring-46.2](#)
gnome-logs: [gnome-logs-45.0](#)
gnome-maps: [gnome-maps-46.11](#)
gnome-menus: [gnome-menus-3.36.0](#)
gnome-nettool: [gnome-nettool-42.0](#)
gnome-online-accounts: [gnome-online-accounts-3.50.4](#)
gnome-power-manager: [gnome-power-manager-43.0](#)
gnome-screenshot: [gnome-screenshot-41.0](#)
gnome-session: [gnome-session-46.0](#)
gnome-settings-daemon: [gnome-settings-daemon-46.0](#)
gnome-shell: [gnome-shell-46.4](#)
gnome-shell-extensions: [gnome-shell-extensions-46.2](#)
gnome-system-monitor: [gnome-system-monitor-46.0](#)
gnome-terminal: [gnome-terminal-3.52.2](#)
gnome-themes-extra: [gnome-themes-extra-3.28](#)
gnome-tweaks: [gnome-tweaks-46.1](#)
gnome-user-docs: [gnome-user-docs-46.4](#)
gnome-weather: [gnome-weather-46.0](#)
Gnumeric: [Gnumeric-1.12.57](#)
GnuPG: [GnuPG-2.4.5](#)
GnuTLS: [GnuTLS-3.8.7.1](#)
GOffice: [GOffice-0.10.57](#)
Gparted: [Gparted-1.6.0](#)
GPGME: [GPGME-1.23.2](#)
GPM: [GPM-1.20.7](#)
gptfdisk: [gptfdisk-1.0.10](#)
grantlee: [Grantlee-5.3.1](#)
graphene: [Graphene-1.10.8](#)
Graphite2: [Graphite2-1.3.14](#)
Graphviz: [Graphviz-12.1.0](#)
Grilo: [Grilo-0.3.16](#)
grub-efi: [GRUB-2.12 for EFI](#)
gsettings-desktop-schemas: [gsettings-desktop-schemas-46.1](#)
Gsl: [gsl-2.8](#)
gsound: [GSound-1.0.3](#)
gspell: [gspell-1.12.2](#)
gst-libav: [gst-libav-1.24.7](#)
gst-plugins-bad: [gst-plugins-bad-1.24.7](#)
gst-plugins-base: [gst-plugins-base-1.24.7](#)
gst-plugins-good: [gst-plugins-good-1.24.7](#)
gst-plugins-ugly: [gst-plugins-ugly-1.24.7](#)
gstreamer: [gstreamer-1.24.7](#)
GTK+3: [GTK+-3.24.43](#)
GTK-Doc: [GTK-Doc-1.34.0](#)
gtk-vnc: [gtk-vnc-1.3.1](#)
GTK-4: [GTK-4.14.5](#)
Gtkmm: [Gtkmm-3.24.9](#)

gtksourceview: [gtksourceview-3.24.11](#)
gtksourceview4: [gtksourceview4-4.8.4](#)
gtksourceview5: [gtksourceview5-5.12.1](#)
Gucharmap: [Gucharmap-15.1.5](#)
Guile: [Guile-3.0.10](#)
Gutenprint: [Gutenprint-5.3.4](#)
Gvfs: [Gvfs-1.54.2](#)
Gwenview: [gwenview-24.08.0](#)
HarfBuzz: [harfBuzz-9.0.0](#)
hatch-fancy-pypi-readme: [Hatch-Fancy-Pypi-Readme-24.1.0](#)
hatchling: [Hatchling-1.25.0](#)
hatch_vcs: [Hatch_vcs-0.4.0](#)
Hdparm: [Hdparm-9.65](#)
Heirloom mailx: [mailx-12.5](#)
Hicolor-icon-theme: [hicolor-icon-theme-0.18](#)
Highlight: [Highlight-4.13](#)
highway: [highway-1.2.0](#)
HTML::Parser: [HTML::Parser-3.83](#)
HTML::Tagset: [HTML::Tagset-3.24](#)
html5lib: [html5lib-1.1](#)
HTTP::CookieJar: [HTTP::CookieJar-0.014](#)
HTTP::Cookies: [HTTP::Cookies-6.11](#)
HTTP::Daemon: [HTTP::Daemon-6.16](#)
HTTP::Date: [HTTP::Date-6.06](#)
HTTP::Message: [HTTP::Message-6.46](#)
HTTP::Negotiate: [HTTP::Negotiate-6.01](#)
ibus: [ibus-1.5.30](#)
icewm: [IceWM-3.6.0](#)
Icon-naming-utils: [icon-naming-utils-0.8.90](#)
icu: [icu-75.1](#)
id3lib: [id3lib-3.8.3](#)
idna: [Idna-3.7](#)
ImageMagick: [ImageMagick-7.1.1-36](#)
imagesize: [Imagesize-1.4.1](#)
imlib2: [imlib2-1.12.3](#)
iniconfig: [Iniconfig-2.0.0](#)
inih: [inih-58](#)
Inkscape: [Inkscape-1.3.2](#)
install-tl-unx: [install-tl-unx](#)
Intel-gmmlib: [Intel-gmmlib-22.4.1](#)
intel-media: [intel-media-24.2.5](#)
intel-vaapi-driver: [intel-vaapi-driver-2.4.1](#)
IO::HTML: [IO::HTML-1.004](#)
IO::Socket::SSL: [IO::Socket::SSL-2.088](#)
IO::String: [IO::String-1.08](#)
IPAEx fonts: [TTF and OTF fonts -- description](#)
IPC::Run3: [IPC::Run3-0.049](#)
IPC::System::Simple: [IPC::System::Simple-1.30](#)
iptables: [iptables-1.8.10](#)
ISO Codes: [ISO Codes-4.16.0](#)
itstool: [itstool-2.0.7](#)
iw: [iw-6.9](#)
jansson: [Jansson-2.14](#)
JasPer: [JasPer-4.2.4](#)
JDK Binary: [Java-22.0.2](#)
jfsutils: [jfsutils-1.1.15](#)
JOE: [JOE-4.6](#)
JSON-C: [JSON-C-0.17](#)
JSON-GLib: [JSON-GLib-1.8.0](#)
K3b: [k3b-24.08.0](#)
Kate: [kate-24.08.0](#)
kcolorpicker: [KColorPicker-0.3.1](#)
kconfig: [Kconfig-6.5.0 for lxqt](#)
KDE Frameworks: [Building KDE Frameworks 6.5.0 \(KF6\).](#)
KDE Plasma: [Building_Plasma](#)
Kdenlive: [kdenlive-24.08.0](#)
kdsoap: [Kdsoap-2.2.0](#)
Kea DHCP Server: [Kea 2.6.1 DHCP Server](#)
keybinder-3.0: [keybinder-3.0-0.3.2](#)
keyutils: [keyutils-1.6.3](#)
khelppcenter: [khelppcenter-24.08.0](#)
kidletime: [kidletime-6.5.0 for lxqt](#)
kimageannotator: [KImageAnnotator-0.7.1](#)
Konsole: [Konsole-24.08.0](#)
konversation: [konversation-24.08.0](#)
kwayland: [kwayland-6.1.4 for lxqt](#)

kwindowsystem: [kwindowsystem-6.5.0 for lxqt](#)
LAME: [LAME-3.100](#)
layer-shell: [layer-shell-qt-6.1.4 for lxqt](#)
ldns: [ldns-1.8.4](#)
Liba52: [Liba52-0.8.0](#)
libadwaita: [libadwaita-1.5.3](#)
libaio: [libaio-0.3.113](#)
Libao: [Libao-1.2.0](#)
libaom: [libaom-3.9.1](#)
libarchive: [libarchive-3.7.4](#)
libass: [libass-0.17.3](#)
libassuan: [libassuan-3.0.1](#)
libatasmart: [libatasmart-0.19](#)
libatomic_ops: [libatomic_ops-7.8.2](#)
libavif: [libavif-1.1.1](#)
libblockdev: [libblockdev-3.1.1](#)
libburn: [libburn-1.5.6](#)
libbytesize: [libbytesize-2.11](#)
libcairomm-1.0: [libcairomm-1.0 \(cairomm-1.14.5\)](#)
libcairomm-1.16: [libcairomm-1.16 \(cairomm-1.18.0\)](#)
libcanberra: [libcanberra-0.30](#)
libcap: [libcap-2.70 with PAM](#)
libcddb: [libcddb-1.3.2](#)
libcdio: [libcdio-2.1.0](#)
libclc: [libclc-18.1.7](#)
libcloudproviders: [libcloudproviders-0.3.6](#)
libcupsfilters: [libcupsfilters-2.0.0](#)
libdaemon: [libdaemon-0.14](#)
libdbusmenu-lxqt: [libdbusmenu-lxqt-0.1.0](#)
libdisplay-info: [libdisplay-info-0.2.0](#)
libdrm: [Libdrm-2.4.122](#)
Libdv: [Libdv-1.0.0](#)
libdvdcss: [libdvdcss-1.4.3](#)
Libdvdnav: [Libdvdnav-6.1.1](#)
Libdvdread: [Libdvdread-6.1.3](#)
libei: [libei-1.3.0](#)
libepoxy: [libepoxy-1.5.10](#)
Liberation fonts: [TTF and OTF fonts -- description](#)
libevdev: [libevdev 1.13.2](#)
libevent: [libevent-2.1.12](#)
libexif: [libexif-0.6.24](#)
libfm-extra: [libfm-extra-1.3.2](#)
libfm-qt: [libfm-qt-2.0.2](#)
libgcrypt: [libgcrypt-1.11.0](#)
libgdata: [libgdata-0.18.1](#)
libgedit-amtk: [libgedit-amtk-5.8.0](#)
libgedit-gtksourceview: [libgedit-gtksourceview-299.2.1](#)
libgee: [libgee-0.20.6](#)
libgpg-error: [libgpg-error-1.50](#)
libgsf: [libgsf-1.14.52](#)
libgtop: [libgtop-2.41.3](#)
libgudev: [libgudev-238](#)
libgusb: [libgusb-0.4.9](#)
libgweather: [libgweather-4.4.2](#)
libgxps: [libgxps-0.3.2](#)
libhandy1: [libhandy-1.8.3](#)
libical: [libical-3.0.18](#)
libidn: [libidn-1.42](#)
libidn2: [libidn2-2.3.7](#)
libinput: [libinput-1.26.1](#)
libisoburn: [libisoburn-1.5.6](#)
libisofs: [libisofs-1.5.6](#)
libjpeg-turbo: [libjpeg-turbo-3.0.1](#)
libjxl: [libjxl-0.10.3](#)
libkcddb: [libkcddb-24.08.0](#)
libkdcraw: [libkdcraw-24.08.0](#)
libkexiv2: [libkexiv2-24.08.0](#)
libksba: [libksba-1.6.7](#)
libkscreen: [libkscreen-6.1.4 for lxqt](#)
liblinear: [liblinear-247](#)
liblxqt: [liblxqt-2.0.0](#)
Libmad: [libmad-0.15.1b](#)
libmbim: [libmbim-1.26.4](#)
Libmng: [libmng-2.0.3](#)
libmnl: [libmnl-1.0.5](#)
libmpeg2: [libmpeg2-0.5.1](#)

libmusicbrainz: [libmusicbrainz-5.1.0](#)
libmypaint: [libmypaint-1.6.1](#)
libndp: [libndp-1.9](#)
libnl: [libnl-3.10.0](#)
libnma: [libnma-1.10.6](#)
libnotify: [libnotify-0.8.3](#)
libnsl: [libnsl-2.0.1](#)
libnvme: [libnvme-1.10](#)
liboauth: [liboauth-1.0.3](#)
Libogg: [libogg-1.3.5](#)
libpaper: [libpaper-2.2.5](#)
libpcap: [libpcap-1.10.4](#)
libpeas: [libpeas-1.36.0](#)
libplacebo: [libplacebo-7.349.0](#)
libpng: [libpng-1.6.43](#)
libportal: [libportal-0.7.1](#)
libppd: [libppd-2.0.0](#)
libpsl: [libpsl-0.21.5](#)
libptytty: [libptytty-2.0](#)
libpwquality: [libpwquality-1.4.5](#)
libqalculate: [libqalculate-5.2.0](#)
libqmi: [libqmi-1.30.8](#)
libqrencode: [qrencode-4.1.1](#)
libqtxdg: [libqtxdg-4.0.0](#)
libraw: [libraw-0.21.2](#)
LibreOffice: [LibreOffice-24.8.0](#)
librsvg: [librsvg-2.58.3](#)
libsamplerate: [libsamplerate-0.2.2](#)
libseccomp: [libseccomp-2.5.5](#)
libsecret: [libsecret-0.21.4](#)
libshumate: [libshumate-1.2.3](#)
libsigc++: [libsigc++-2.12.1](#)
libsigc++-3: [libsigc++-3.6.0](#)
libsigsegv: [libsigsegv-2.14](#)
libslirp: [libslirp-4.8.0](#)
libsndfile: [libsndfile-1.2.2](#)
libsoup: [libsoup-2.74.3](#)
libsoup3: [libsoup-3.4.4](#)
libspiro: [Libspiro-20220722](#)
libssh2: [libssh2-1.11.0](#)
libstatgrab: [libstatgrab-0.92.1](#)
libsysstat: [libsysstat-1.0.0](#)
libtasn1: [libtasn1-4.19.0](#)
libtiff: [libtiff-4.6.0](#)
libtirpc: [libtirpc-1.3.5](#)
libunistring: [libunistring-1.2](#)
libunwind: [libunwind-1.6.2](#)
liburcu: [liburcu-0.14.0](#)
libusb: [libusb-1.0.27](#)
libuv: [libuv-1.48.0](#)
libva: [libva-2.22.0](#)
libvdpau: [libvdpau-1.5](#)
libvdpau-va-gl: [libvdpau-va-gl-0.4.2](#)
Libvorbis: [libvorbis-1.3.7](#)
libvpx: [libvpx-1.14.1](#)
libwacom: [libwacom-2.12.2](#)
libwebp: [libwebp-1.4.0](#)
libwnck 3: [libwnck-43.0](#)
libwww-perl: [LWP_\(libwww-perl-6.77\)](#)
libXau: [libXau-1.0.11](#)
libxcb: [libxcb-1.17.0](#)
libxcvt: [libxcvt-0.1.2](#)
libXdmcp: [libXdmcp-1.1.5](#)
libxfc4ui: [libxfc4ui-4.18.6](#)
libxfc4util: [libxfc4util-4.18.2](#)
libxkbcommon: [libxkbcommon-1.7.0](#)
libxklavier: [libxklavier-5.4](#)
libxml2: [libxml2-2.13.3](#)
libxmlb: [libxmlb-0.3.19](#)
libxslt: [libxslt-1.1.42](#)
libyaml: [libyaml-0.2.5](#)
lightdm: [lightdm-1.32.0](#)
Lingua::Translit: [Lingua::Translit-0.29](#)
Links: [Links-2.30](#)
Linux-PAM: [Linux-PAM-1.6.1](#)
List::AllUtils: [List::AllUtils-0.19](#)

List::MoreUtils: [List::MoreUtils-0.430](#)
List::MoreUtils::XS: [List::MoreUtils::XS-0.430](#)
List::SomeUtils: [List::SomeUtils-0.59](#)
List::SomeUtils::XS: [List::SomeUtils::XS-0.58](#)
List::UtilsBy: [List::UtilsBy-0.12](#)
Little CMS2: [Little CMS-2.16](#)
LLVM: [LLVM-18.1.7](#)
Imdb: [Imdb-0.9.31](#)
Im_sensors: [Im-sensors-3-6-0](#)
Log::Log4perl: [Log::Log4perl-1.57](#)
log4cplus: [log4cplus-2.1.1](#)
logrotate: [Logrotate-3.22.0](#)
Lohit fonts: [TTF and OTF fonts -- description](#)
LSB-Tools: [LSB-Tools-0.12](#)
Isof: [Isof-4.99.0](#)
Lua: [Lua-5.4.7](#)
[Lua-5.2.4](#)
Iuit: [Iuit-20240102](#)
LVM2: [LVM2-2.03.26](#)
LWP::MediaTypes: [LWP::MediaTypes-6.04](#)
LWP::Protocol::https: [LWP::Protocol::https-6.14](#)
lxde-icon-theme: [lxde-icon-theme-0.5.1](#)
lximage-qt: [lximage-qt-2.0.1](#)
lxml: [lxml-5.3.0](#)
lxqt-about: [lxqt-about-2.0.0](#)
lxqt-admin: [lxqt-admin-2.0.0](#)
lxqt-archiver: [lxqt-archiver-1.0.0](#)
lxqt-build-tools: [lxqt-build-tools-2.0.0](#)
lxqt-config: [lxqt-config-2.0.0](#)
lxqt-globalkeys: [lxqt-globalkeys-2.0.0](#)
lxqt-menu-data: [lxqt-menu-data-2.0.0](#)
lxqt-notificationd: [lxqt-notificationd-2.0.1](#)
lxqt.openssh-askpass: [lxqt.openssh-askpass-2.0.1](#)
lxqt-panel: [lxqt-panel-2.0.1](#)
lxqt-policykit: [lxqt-policykit-2.0.0](#)
lxqt-powermanagement: [lxqt-powermanagement-2.0.0](#)
lxqt-qtpugin: [lxqt-qtpugin-2.0.0](#)
lxqt-runner: [lxqt-runner-2.0.0](#)
lxqt-session: [lxqt-session-2.0.0](#)
lxqt-sudo: [lxqt-sudo-2.0.0](#)
lxqt-themes: [lxqt-themes-2.0.0](#)
Lynx: [Lynx-2.9.2](#)
LZO: [LZO-2.10](#)
make-ca: [make-ca-1.14](#)
Mako: [Mako-1.3.5](#)
mandoc: [mandoc-1.14.6](#)
MariaDB: [MariaDB-10.11.8](#)
Markdown: [Markdown-3.6](#)
MC: [MC-4.8.32](#)
mdadm: [mdadm-4.3](#)
menu-cache: [menu-cache-1.1.0](#)
mercurial: [Mercurial-6.8.1](#)
Mesa: [Mesa-24.1.5](#)
meson_python: [Meson_python-0.16.0](#)
MIME::Charset: [MIME::Charset-1.013.1](#)
MIT Kerberos V5: [MIT Kerberos V5-1.21.3](#)
mlt: [mlt-7.26.0](#)
ModemManager: [ModemManager-1.18.12](#)
Module::Build: [Module::Build-0.4234](#)
Module::Implementation: [Module::Implementation-0.09](#)
Module::Runtime: [Module::Runtime-0.016](#)
Mousepad: [Mousepad-0.6.2](#)
mpg123: [mpg123-1.32.7](#)
mpv: [mpv-0.38.0](#)
MRO::Compat: [MRO::Compat-0.15](#)
msgpack: [Msgpack-1.0.8](#)
mtdev: [mtdev-1.1.7](#)
muparser: [muparser-2.3.4](#)
mupdf: [MuPDF-1.24.8](#)
mutt: [Mutt-2.2.13](#)
Mutter: [Mutter-46.4](#)
mypaint-brushes: [mypaint-brushes-1.3.1](#)
namespace::autoclean: [namespace::autoclean-0.29](#)
namespace::clean: [namespace::clean-0.27](#)
Nano: [Nano-8.1](#)
NASM: [NASM-2.16.03](#)

Nautilus: [Nautilus-46.2](#)
NcFTP: [NcFTP-3.2.7](#)
neon: [neon-0.33.0](#)
Net::DNS: [Net::DNS-1.46](#)
Net::HTTP: [Net::HTTP-6.23](#)
Net::SSLeay: [Net::SSLeay-1.94](#)
Net-tools: [Net-tools-2.10](#)
Nettle: [Nettle-3.10](#)
network-manager-applet: [network-manager-applet-1.34.0](#)
NetworkManager: [NetworkManager-1.48.8](#)
newt: [newt-0.52.24](#)
NFS Utilities: [NFS-Utils-2.6.4](#)
nghttp2: [nghttp2-1.62.1](#)
Nmap: [Nmap-7.95](#)
nodejs: [Node.js-20.16.0](#)
notification-daemon: [notification-daemon-3.20.0](#)
Noto fonts: [TTF and OTF fonts -- description](#)
NPth: [npth-1.7](#)
NSPR: [NSPR-4.35](#)
NSS: [NSS-3.103](#)
ntfs-3g: [ntfs-3g-2022.10.3](#)
ntp: [ntp-4.2.8p18](#)
Number::Compare: [Number::Compare-0.03](#)
numpy: [NumPy-2.1.0](#)
obconf-qt: [obconf-qt-0.16.4](#)
Okular: [okular-24.08.0](#)
Openbox: [openbox-3.6.1](#)
opencv: [opencv-4.10.0](#)
OpenJade: [OpenJade-1.3.2](#)
OpenJDK: [OpenJDK-22.0.2](#)
OpenJPEG: [OpenJPEG-2.5.2](#)
OpenLDAP: [OpenLDAP-2.6.8](#)
OpenSP: [OpenSP-1.5.2](#)
OpenSSH: [OpenSSH-9.8p1](#)
Opus: [Opus-1.5.2](#)
oxygen-icons: [oxygen-icons-6.0.0](#)
p11-kit: [p11-kit-0.25.5](#)
p7zip: [p7zip-17.04](#)
Package::Stash: [Package::Stash-0.40](#)
packaging: [Packaging-24.1](#)
Pango: [Pango-1.54.0](#)
Pangomm: [Pangomm-2.46.4](#)
Pangomm: [Pangomm-2.54.0](#)
paps: [paps-0.8.0](#)
Params::Validate: [Params::Validate-1.31](#)
Params::ValidationCompiler: [Params::ValidationCompiler-0.31](#)
Parole: [Parole-4.18.1](#)
Parse::RecDescent: [Parse::RecDescent-1.967015](#)
Parse::Yapp: [Parse::Yapp-1.21](#)
Parted: [parted-3.6](#)
patchelf: [Patchelf-0.18.0](#)
Path::Tiny: [Path::Tiny-0.146](#)
pathspec: [Pathspec-0.12.1](#)
pavucontrol: [pavucontrol-6.1](#)
pavucontrol-qt: [pavucontrol-qt-2.0.0](#)
pax: [Pax-20240817](#)
pciutils: [pciutils-3.13.0](#)
pcmanfm-qt: [pcmanfm-qt-2.0.0](#)
PCRE2: [PCRE2-10.44](#)
Perl Module Dependencies: [Perl Module Dependencies](#)
Perl Modules: [Perl Modules](#)
PerlIO::utf8_strict: [PerlIO::utf8_strict-0.010](#)
phonon: [Phonon-4.12.0](#)
phonon-backend-vlc: [Phonon-backend-vlc-0.12.0](#)
PHP: [PHP-8.3.10](#)
PIN-Entry: [pinentry-1.3.1](#)
pipewire: [Pipewire-1.2.3](#)
Pixman: [Pixman-0.43.4](#)
plasma-activities: [plasma-activities-6.1.4](#)
plasma-wayland-protocols: [plasma-wayland-protocols-1.13.0](#)
pluggy: [Pluggy-1.5.0](#)
ply: [ply-3.11](#)
pm-utils: [pm-utils-1.4.1](#)
pnmixer: [pnmixer-0.7.2](#)
Polkit: [Polkit-125](#)
polkit-gnome: [polkit-gnome-0.105](#)

polkit-qt: [Polkit-Qt-0.200.0](#)
Poppler: [Poppler-24.08.0](#)
Popt: [Popt-1.19](#)
Postfix: [Postfix-3.9.0](#)
PostgreSQL: [PostgreSQL-16.4](#)
Potrace: [Potrace-1.16](#)
power-profiles-daemon: [Power-profiles-daemon-0.21](#)
Procmail: [Procmail-3.24](#)
Proftpd: [ProFTPD-1.3.8b](#)
Protobuf: [Protobuf-27.3](#)
Protobuf-c: [Protobuf-c-1.5.0](#)
psf-tools: [About Console Fonts -- description](#)
psutil: [psutil-6.0.0](#)
PulseAudio: [PulseAudio-17.0](#)
pulseaudio-qt: [pulseaudio-qt-1.5.0](#)
Py3c: [Py3c-1.4](#)
PyAtSpi2: [PyAtSpi2-2.46.1](#)
PyCairo: [PyCairo-1.26.1](#)
pygdbmi: [pygdbmi-0.11.0.0](#)
Pygments: [Pygments-2.18.0](#)
PyGObject3: [PyGObject-3.48.2](#)
pyparsing: [pyparsing-3.1.2](#)
pyproject-metadata: [Pyproject-Metadata-0.8.0](#)
pyserial: [pySerial-3.5](#)
pytest: [Pytest-8.3.2](#)
Python Dependencies: [Python Dependencies](#)
Python Modules: [Python Modules](#)
Python3: [Python-3.12.5](#)
Python3.11: [Python-3.11.1](#)
pytz: [Pytz-2024.1](#)
PyXDG: [PyXDG-0.28](#)
PyYAML: [PyYAML-6.0.2](#)
qca: [Qca-2.3.9](#)
qemu: [qemu-9.0.2](#)
Qpdf: [Qpdf-11.9.1](#)
qps: [qps-2.9.0](#)
Qt5 Components: [qt5-components-5.15.14](#)
Qt6: [Qt-6.7.2](#)
qterminal: [qterminal-2.0.1](#)
qtermwidget: [qtermwidget-2.0.1](#)
qtwebengine: [QtWebEngine-6.7.2](#)
qtxdg-tools: [qtxdg-tools-4.0.0](#)
Raptor: [Raptor-2.0.16](#)
Rasqal: [Rasqal-0.9.33](#)
recommonmark: [Recommonmark-0.7.1](#)
Redland: [Redland-1.0.17](#)
Regexp::Common: [Regexp::Common-2024080801](#)
requests: [Requests-2.32.3](#)
rest: [rest-0.9.1](#)
Ristretto: [Ristretto-0.13.2](#)
Role::Tiny: [Role::Tiny-2.002004](#)
rpcbind: [rpcbind-1.2.7](#)
rpcsvc-proto: [rpcsvc-proto-1.4.4](#)
rsync: [rsync-3.3.0](#)
Ruby: [Ruby-3.3.4](#)
Rust: [Rustc-1.80.1](#)
rust-bindgen: [rust-bindgen-0.70.0](#)
rxvt-unicode: [rxvt-unicode-9.31](#)
Samba: [Samba-4.20.4](#)
SANE: [SANE-1.2.1](#)
sassc: [sassc-3.6.2](#)
SBC: [SBC-2.0](#)
SCons: [SCons-4.8.0](#)
Scope::Guard: [Scope::Guard-0.21](#)
scour: [Scour-0.38.2](#)
Screen: [Screen-4.9.1](#)
screengrab: [screengrab-2.8.0](#)
sddm: [sddm-0.21.0](#)
SDL: [sdl12-compat-1.2.68](#)
SDL2: [SDL2-2.30.6](#)
Seahorse: [Seahorse-43.0](#)
seamonkey: [Seamonkey-2.53.18.2](#)
sendmail: [sendmail-8.18.1](#)
sentry-sdk: [sentry-sdk-2.13.0](#)
Serf: [Serf-1.3.10](#)
setup-tools_scm: [Setup-tools_scm-8.1.0](#)

sg3_utils: [sg3_utils-1.48](#)
SGML Common: [sgml-common-0.6.3](#)
SGMLSpm: [SGMLSpm-1.1](#)
Shadow: [Shadow-4.16.0](#)
shared-mime-info: [shared-mime-info-2.4](#)
Sharutils: [Sharutils-4.15.2](#)
Simple-scan: [Simple-scan-46.0](#)
six: [six-1.16.0](#)
slang: [slang-2.3.3](#)
smartmontools: [smartmontools-7.4](#)
smartypants: [Smartypants-2.0.1](#)
Snapshot: [Snapshot-46.3](#)
snowballstemmer: [Snowballstemmer-2.2.0](#)
solid: [solid-6.5.0 for lxqt](#)
Sort::Key: [Sort::Key-1.33](#)
sound-theme-freedesktop: [sound-theme-freedesktop-0.8](#)
SoundTouch: [SoundTouch-2.3.3](#)
Source Code Pro: [TTF and OTF fonts -- description](#)
Specio: [Specio-0.48](#)
Speex: [Speex-1.2.1](#)
sphinx: [Sphinx-8.0.2](#)
sphinxcontrib-applehelp: [Sphinxcontrib-applehelp-2.0.0](#)
sphinxcontrib-devhelp: [Sphinxcontrib-devhelp-2.0.0](#)
sphinxcontrib-htmlhelp: [Sphinxcontrib-htmlhelp-2.0.1](#)
sphinxcontrib-jquery: [Sphinxcontrib-jquery-4.1](#)
sphinxcontrib-jsmath: [Sphinxcontrib-jsmath-1.0.1](#)
sphinxcontrib-qthelp: [Sphinxcontrib-qthelp-2.0.0](#)
sphinxcontrib-serializinghtml: [Sphinxcontrib-serializinghtml-2.0.0](#)
sphinx_rtd_theme: [Sphinx_rtd_theme-2.0.0](#)
SpiderMonkey: [SpiderMonkey from firefox-115.14.0](#)
SPIRV-Headers: [SPIRV-Headers-1.3.290.0](#)
SPIRV-LLVM-Translator: [SPIRV-LLVM-Translator-18.1.3](#)
SPIRV-Tools: [SPIRV-Tools-1.3.290.0](#)
SQLite: [SQLite-3.46.1](#)
ssh-askpass: [ssh-askpass-9.8p1](#)
sshfs: [sshfs-3.7.3](#)
startup-notification: [startup-notification-0.12](#)
stunnel: [stunnel-5.72](#)
Sub::Exporter::Progressive: [Sub::Exporter::Progressive-0.001013](#)
Sub::Identify: [Sub::Identify-0.14](#)
Sub::Quote: [Sub::Quote-2.006008](#)
Sub::Uplevel: [Sub::Uplevel-0.2800](#)
Subversion: [Subversion-1.14.3](#)
Sudo: [Sudo-1.9.15p5](#)
SWIG: [SWIG-4.2.1](#)
sysmon-qt: [sysmon-qt-2.0.1](#)
Sysstat: [Sysstat-12.7.6](#)
systemd: [Systemd-256.4](#)
Taglib: [Taglib-2.0.1](#)
Talloc: [Talloc-2.4.2](#)
Tcsh: [Tcsh-6.24.13](#)
Tecla: [Tecla-46.0](#)
Term::Table: [Term::Table-0.022](#)
terminus-font: [About Console Fonts -- description](#)
Test::Command: [Test::Command-0.11](#)
Test::Deep: [Test::Deep-1.204](#)
Test::Differences: [Test::Differences-0.71](#)
Test::Exception: [Test::Exception-0.43](#)
Test::Fatal: [Test::Fatal-0.017](#)
Test::File: [Test::File-1.993](#)
Test::File::ShareDir: [Test::File::ShareDir-1.001002](#)
Test::LeakTrace: [Test::LeakTrace-0.17](#)
Test::Needs: [Test::Needs-0.002010](#)
Test::Requires: [Test::Requires-0.11](#)
Test::RequiresInternet: [Test::RequiresInternet-0.05](#)
Test::Simple: [Test::Simple-1.302200](#)
Test::utf8: [Test::utf8-1.02](#)
Test::Warnings: [Test::Warnings-0.033](#)
Test::Without::Module: [Test::Without::Module-0.23](#)
Test2::Plugin::NoWarnings: [Test2::Plugin::NoWarnings-0.10](#)
texlive: [texlive-20240312-source](#)
Text::BibTeX: [Text::BibTeX-0.89](#)
Text::CSV: [Text::CSV-2.04](#)
Text::CSV_XS: [Text::CSV_XS-1.56](#)
Text::Diff: [Text::Diff-1.45](#)
Text::Glob: [Text::Glob-0.11](#)

Text::Roman: [Text::Roman-3.5](#)
Thunar: [thunar-4.18.11](#)
thunar-volman: [thunar-volman-4.18.0](#)
Thunderbird: [Thunderbird-128.1.0esr](#)
Tidy HTML5: [tidy-html5-5.8.0](#)
Tie::Cycle: [Tie::Cycle-1.228](#)
tigervnc: [Tigervnc-1.14.0](#)
time: [Time-1.9](#)
TimeDate: [TimeDate-2.33](#)
Tk: [Tk-8.6.14](#)
totem-pl-parser: [totem-pl-parser-3.26.6](#)
Traceroute: [Traceroute-2.1.5](#)
Tracker3: [Tracker-3.7.3](#)
tracker3-miners: [Tracker-miners-3.7.3](#)
Transmission: [Transmission-4.0.6](#)
tree: [tree-2.1.3](#)
Tripwire: [Tripwire-2.4.3.7](#)
trove-classifiers: [Trove-Classifiers-2024.7.2](#)
Try::Tiny: [Try::Tiny-0.31](#)
TTF and OTF fonts: [TTF and OTF fonts](#)
Tumbler: [tumbler-4.18.2](#)
twm: [twm-1.0.12](#)
typogrify: [Typogrify-2.0.7](#)
uchardet: [Uchardet-0.0.8](#)
UDisks2: [UDisks-2.10.1](#)
uhttpmock: [uhttpmock-0.11.0](#)
umockdev: [Umockdev-0.18.3](#)
Unbound: [Unbound-1.21.0](#)
Unicode::Collate: [Unicode::Collate-1.31](#)
Unicode::LineBreak: [Unicode::LineBreak-2019.001](#)
unifdef: [unifdef-2.12](#)
unixODBC: [unixODBC-2.3.12](#)
UnRar: [UnRar-7.0.9](#)
UnZip: [UnZip-6.0](#)
UPower: [UPower-1.90.4](#)
URI: [URI-5.28](#)
urllib3: [Urllib3-2.2.2](#)
usbutils: [usbutils-017](#)
utfcpp: [utfcpp-4.0.5](#)
util-macros: [util-macros-1.20.1](#)
v4l-utils: [v4l-utils-1.28.1](#)
Vala: [Vala-0.56.17](#)
Valgrind: [Valgrind-3.23.0](#)
Variable::Magic: [Variable::Magic-0.64](#)
Vim: [Vim-9.1.0660](#)
VLC: [VLC-3.0.21](#)
VL Gothic: [TTF and OTF fonts -- description](#)
Vorbis Tools: [vorbis-tools-1.4.2](#)
VTE: [VTE-0.76.4](#)
Vulkan-Headers: [Vulkan-Headers-1.3.294](#)
Vulkan-Loader: [Vulkan-Loader-1.3.294](#)
Wayland: [Wayland-1.23.0](#)
wayland-protocols: [Wayland-Protocols-1.36](#)
webencodings: [webencodings-0.5.1](#)
WebKitGTK: [WebKitGTK-2.44.3](#)
webp-pixbuf-loader: [webp-pixbuf-loader-0.2.7](#)
WenQuanYi Zen Hei: [TTF and OTF fonts -- description](#)
Wget: [Wget-1.24.5](#)
Which: [Which-2.21_and_Alternatives](#)
Whois: [Whois-5.4.3](#)
Wireless Tools: [Wireless Tools-29](#)
Wireplumber: [Wireplumber-0.5.5](#)
Wireshark: [Wireshark-4.2.6](#)
woff2: [woff2-1.0.2](#)
wpa_supplicant: [wpa_supplicant-2.11](#)
Wv: [wv-1.2.9](#)
WWW::RobotRules: [WWW::RobotRules-6.02](#)
x264: [x264-20240812](#)
x265: [x265-3.6](#)
xapian: [Xapian-1.4.26](#)
xarchiver: [xarchiver-0.5.4.23](#)
xbitmaps: [xbitmaps-1.1.3](#)
xcb-proto: [xcb-proto-1.17.0](#)
xcb-util: [xcb-util-0.4.1](#)
xcb-util-cursor: [xcb-util-cursor-0.1.4](#)
xcb-util-image: [xcb-util-image-0.4.1](#)

xcb-util-keysyms: [xcb-util-keysyms-0.4.1](#)
xcb-util-renderutil: [xcb-util-renderutil-0.3.10](#)
xcb-util-wm: [xcb-util-wm-0.4.2](#)
xclock: [xclock-1.1.1](#)
xcursor-themes: [xcursor-themes-1.0.7](#)
xdg-dbus-proxy: [xdg-dbus-proxy-0.1.5](#)
xdg-desktop-portal: [xdg-desktop-portal-1.18.2](#)
xdg-desktop-portal-gnome: [xdg-desktop-portal-gnome-46.2](#)
xdg-desktop-portal-gtk: [xdg-desktop-portal-gtk-1.15.1](#)
xdg-desktop-portal-lxqt: [xdg-desktop-portal-lxqt-1.0.2](#)
xdg-user-dirs: [Xdg-user-dirs-0.18](#)
xdg-utils: [xdg-utils-1.2.1](#)
xdotool: [xdotool-3.20211022.1](#)
Xfburn: [Xfburn-0.7.2](#)
xfce4-appfinder: [xfce4-appfinder-4.18.1](#)
xfce4-dev-tools: [xfce4-dev-tools-4.18.1](#)
xfce4-notifyd: [xfce4-notifyd-0.9.6](#)
xfce4-panel: [xfce4-panel-4.18.6](#)
xfce4-power-manager: [xfce4-power-manager-4.18.4](#)
xfce4-pulseaudio-plugin: [xfce4-pulseaudio-plugin-0.4.8](#)
xfce4-session: [xfce4-session-4.18.4](#)
xfce4-settings: [xfce4-settings-4.18.6](#)
xfce4-terminal: [xfce4-terminal-1.1.3](#)
Xfconf: [Xfconf-4.18.3](#)
Xfdesktop: [Xfdesktop-4.18.1](#)
xfsprogs: [xfsprogs-6.9.0](#)
Xfwm4: [Xfwm4-4.18.0](#)
xindy: [xindy-2.5.1](#)
Xine Libraries: [xine-lib-1.2.13](#)
Xine User Interface: [xine-ui-0.99.14](#)
xinit: [xinit-1.4.2](#)
XKeyboardConfig: [XKeyboardConfig-2.42](#)
XML::LibXML: [XML::LibXML-2.0210](#)
XML::LibXML::Simple: [XML::LibXML::Simple-1.01](#)
XML-LibXSLT: [XML::LibXSLT-2.003000](#)
XML::NamespaceSupport: [XML::NamespaceSupport-1.12](#)
XML::SAX: [XML::SAX-1.02](#)
XML::SAX::Base: [XML::SAX::Base-1.09](#)
XML::Simple: [XML::Simple-2.25](#)
XML::Writer: [XML::Writer-0.900](#)
xmlito: [xmlito-0.0.29](#)
xorg-evdev-driver: [Xorg_Evdev_Driver-2.10.6](#)
xorg-libinput-driver: [Xorg_Libinput_Driver-1.4.0](#)
xorg-server: [Xorg_Server-21.1.13](#)
xorg-synaptics-driver: [Xorg_Synaptics_Driver-1.9.2](#)
xorg-wacom-driver: [Xorg_Wacom_Driver-1.2.2](#)
Xorg: [Introduction_to_Xorg-7](#)
Xorg-7 Applications: [Xorg_Applications](#)
Xorg Fonts: [Xorg_Fonts](#)
xorg7-input-driver: [Xorg_Input_Drivers](#)
Xorg Legacy: [Xorg_Legacy](#)
Xorg Libraries: [Xorg_Libraries](#)
xorgproto: [xorgproto-2024.1](#)
XScreenSaver: [XScreenSaver-6.09](#)
xterm: [xterm-393](#)
XviD: [XviD-1.3.7](#)
xwayland: [Xwayland-24.1.2](#)
yasm: [yasm-1.3.0](#)
Yelp: [Yelp-42.2](#)
yelp-xsl: [yelp-xsl-42.1](#)
Zip: [Zip-3.0](#)
zsh: [zsh-5.9](#)

Programs

7z: [p7zip-17.04 -- description](#)
7za: [p7zip-17.04 -- description](#)
7zr: [p7zip-17.04 -- description](#)
a2x: [Asciidoc-10.2.1 -- description](#)
a52dec: [LibA52-0.8.0 -- description](#)
aafire: [AAlib-1.4rc5 -- description](#)
aainfo: [AAlib-1.4rc5 -- description](#)
aalib-config: [AAlib-1.4rc5 -- description](#)
aasavefont: [AAlib-1.4rc5 -- description](#)
aatest: [AAlib-1.4rc5 -- description](#)
aaxine: [xine-ui-0.99.14 -- description](#)

avahi-publish-service: [Avahi-0.8](#) -- [description](#)
avahi-resolve: [Avahi-0.8](#) -- [description](#)
avahi-resolve-address: [Avahi-0.8](#) -- [description](#)
avahi-resolve-host-name: [Avahi-0.8](#) -- [description](#)
avahi-set-host-name: [Avahi-0.8](#) -- [description](#)
axfer: [alsa-utils-1.2.12](#) -- [description](#)
balsa: [Balsa-2.6.4](#) -- [description](#)
balsa-ab: [Balsa-2.6.4](#) -- [description](#)
balsam: [Qt-6.7.2](#) -- [description](#)
baobab: [Baobab-46.0](#) -- [description](#)
batch: [at-3.2.5](#) -- [description](#)
bcomp: [Graphviz-12.1.0](#) -- [description](#)
bdftopcf: [Xorg_Legacy](#) -- [description](#)
bftruncate: [Xorg_Fonts](#) -- [description](#)
benchmark_xl: [libjxl-0.10.3](#) -- [description](#)
bf_compact: [Bogofilter-1.2.5](#) -- [description](#)
bf_copy: [Bogofilter-1.2.5](#) -- [description](#)
bf_tar: [Bogofilter-1.2.5](#) -- [description](#)
biber: [Biber-2.20](#) -- [description](#)
biblex: [Text::BibTeX-0.89](#) -- [description](#)
bibparse: [Text::BibTeX-0.89](#) -- [description](#)
bindgen: [rust-bindgen-0.70.0](#) -- [description](#)
blkdeactivate: [LVM-2.03.26](#) -- [description](#)
blocks: [FLTK-1.3.9](#) -- [description](#)
bluefish: [Bluefish-2.2.15](#) -- [description](#)
bluemoon: [BlueZ-5.77](#) -- [description](#)
bluetooth-sendto: [gnome-bluetooth-46.1](#) -- [description](#)
bluetoothctl: [BlueZ-5.77](#) -- [description](#)
bluetoothd: [BlueZ-5.77](#) -- [description](#)
bogofilter: [Bogofilter-1.2.5](#) -- [description](#)
bogolexer: [Bogofilter-1.2.5](#) -- [description](#)
bogotune: [Bogofilter-1.2.5](#) -- [description](#)
bogoupgrade: [Bogofilter-1.2.5](#) -- [description](#)
bogoutil: [Bogofilter-1.2.5](#) -- [description](#)
brasero: [Brasero-3.12.3](#) -- [description](#)
brctl: [bridge-utils-1.7.1](#) -- [description](#)
broadwayd: [GTK+-3.24.43](#) -- [description](#)
brotli: [brotli-1.1.0](#) -- [description](#)
bscalc: [libbytesize-2.11](#) -- [description](#)
bsdcat: [libarchive-3.7.4](#) -- [description](#)
bsdcpio: [libarchive-3.7.4](#) -- [description](#)
bsdstar: [libarchive-3.7.4](#) -- [description](#)
bsdunzip: [libarchive-3.7.4](#) -- [description](#)
bssh: [Avahi-0.8](#) -- [description](#)
btcflash: [Cdrtools-3.02a09](#) -- [description](#)
btmon: [BlueZ-5.77](#) -- [description](#)
btrfs: [Btrfs-6.10.1](#) -- [description](#)
btrfs-convert: [Btrfs-6.10.1](#) -- [description](#)
btrfs-find-root: [Btrfs-6.10.1](#) -- [description](#)
btrfs-map-logical: [Btrfs-6.10.1](#) -- [description](#)
btrfs-select-super: [Btrfs-6.10.1](#) -- [description](#)
btrfstune: [Btrfs-6.10.1](#) -- [description](#)
bugpoint: [LLVM-18.1.7](#) -- [description](#)
bundle: [Ruby-3.3.4](#) -- [description](#)
bundler: [Ruby-3.3.4](#) -- [description](#)
bvnc: [Avahi-0.8](#) -- [description](#)
bwrap: [Bubblewrap-0.9.0](#) -- [description](#)
c-index-test: [LLVM-18.1.7](#) -- [description](#)
cacaxine: [xine-ui-0.99.14](#) -- [description](#)
doesitcache: [CacheControl-0.14.0](#)
cairo-trace: [Cairo-1.18.0](#) -- [description](#)
calc_tickadj: [ntp-4.2.8p18](#) -- [description](#)
callgrind_annotate: [Valgrind-3.23.0](#) -- [description](#)
callgrind_control: [Valgrind-3.23.0](#) -- [description](#)
canberra-gtk-play: [libcanberra-0.30](#) -- [description](#)
canbusutil: [Qt-6.7.2](#) -- [description](#)
cancel: [Cups-2.4.10](#) -- [description](#)
capinfos: [Wireshark-4.2.6](#) -- [description](#)
captcha: [Wireshark-4.2.6](#) -- [description](#)
cargo: [Rustc-1.80.1](#) -- [description](#)
cargo-clippy: [Rustc-1.80.1](#) -- [description](#)
cargo-fmt: [Rustc-1.80.1](#) -- [description](#)
cbindgen: [Cbindgen-0.27.0](#) -- [description](#)
ccache-swig: [SWIG-4.2.1](#) -- [description](#)
ccmake: [CMake-3.30.2](#) -- [description](#)
ccomps: [Graphviz-12.1.0](#) -- [description](#)

cd-convert: [colord-gtk-0.3.1](#) -- [description](#)
cd-create-profile: [Colord-1.4.7](#) -- [description](#)
cd-drive: [libcdio-2.1.0](#) -- [description](#)
cd-fix-profile: [Colord-1.4.7](#) -- [description](#)
cd-iccdump: [Colord-1.4.7](#) -- [description](#)
cd-info: [libcdio-2.1.0](#) -- [description](#)
cd-it8: [Colord-1.4.7](#) -- [description](#)
cd-paranoia: [libcdio-2.1.0](#) -- [description](#)
cd-read: [libcdio-2.1.0](#) -- [description](#)
cdda-player: [libcdio-2.1.0](#) -- [description](#)
cdda2wav: [Cdrtools-3.02a09](#) -- [description](#)
cddb_query: [libcddb-1.3.2](#) -- [description](#)
cdparanoia: [CDParanoia-III-10.2](#) -- [description](#)
cdrdao: [Cdrdao-1.2.4](#) -- [description](#)
cdrecord: [Cdrtools-3.02a09](#) -- [description](#)
cdrskin: [libburn-1.5.6](#) -- [description](#)
certtool: [GnuTLS-3.8.7.1](#) -- [description](#)
certutil: [NSS-3.103](#) -- [description](#)
cgdisk: [gptfdisk-1.0.10](#) -- [description](#)
cg_annotate: [Valgrind-3.23.0](#) -- [description](#)
cg_diff: [Valgrind-3.23.0](#) -- [description](#)
cg_merge: [Valgrind-3.23.0](#) -- [description](#)
chardetect: [Chardet-5.2.0](#) -- [description](#)
checkers: [FLTK-1.3.9](#) -- [description](#)
checkgid: [Apache-2.4.62](#) -- [description](#)
checkXML6: [Building KDE Frameworks 6.5.0 \(KF6\)](#) -- [description](#)
cifs.idmap: [cifs-utils-7.0](#) -- [description](#)
cifs.upcall: [cifs-utils-7.0](#) -- [description](#)
cifscreds: [cifs-utils-7.0](#) -- [description](#)
cifsdd: [Samba-4.20.4](#) -- [description](#)
cifsiostat: [Sysstat-12.7.6](#) -- [description](#)
circo: [Graphviz-12.1.0](#) -- [description](#)
cjpeg: [libjpeg-turbo-3.0.1](#) -- [description](#)
cjxl: [libjxl-0.10.3](#) -- [description](#)
clang: [LLVM-18.1.7](#) -- [description](#)
clang-check: [LLVM-18.1.7](#) -- [description](#)
clang-extdef-mapping: [LLVM-18.1.7](#) -- [description](#)
clang-format: [LLVM-18.1.7](#) -- [description](#)
clang-linker-wrapper: [LLVM-18.1.7](#) -- [description](#)
clang-offload-bundler: [LLVM-18.1.7](#) -- [description](#)
clang-offload-packager: [LLVM-18.1.7](#) -- [description](#)
clang-refactor: [LLVM-18.1.7](#) -- [description](#)
clang-rename: [LLVM-18.1.7](#) -- [description](#)
clang-scan-deps: [LLVM-18.1.7](#) -- [description](#)
clang-tblgen: [LLVM-18.1.7](#) -- [description](#)
clippy-driver: [Rustc-1.80.1](#) -- [description](#)
clisp: [Clisp-2.49](#) -- [description](#)
clisp-link: [Clisp-2.49](#) -- [description](#)
cluster: [Graphviz-12.1.0](#) -- [description](#)
clusterdb: [PostgreSQL-16.4](#) -- [description](#)
cm2html: [Recommomark-0.7.1](#) -- [description](#)
cm2latex: [Recommomark-0.7.1](#) -- [description](#)
cm2man: [Recommomark-0.7.1](#) -- [description](#)
cm2pseudoxml: [Recommomark-0.7.1](#) -- [description](#)
cm2xetex: [Recommomark-0.7.1](#) -- [description](#)
cm2xml: [Recommomark-0.7.1](#) -- [description](#)
cmake: [CMake-3.30.2](#) -- [description](#)
cmake-gui: [CMake-3.30.2](#) -- [description](#)
cmark: [Commonmark-0.9.1](#) -- [description](#)
collateindex.pl: [docbook-dsssl-1.79](#) -- [description](#)
colormgr: [Colord-1.4.7](#) -- [description](#)
compare: [ImageMagick-7.1.1-36](#) -- [description](#)
comparerender: [Graphite2-1.3.14](#) -- [description](#)
compface: [Compface-1.5.2](#) -- [description](#)
complete-ant-cmd.pl: [apache-ant-1.10.14](#) -- [description](#)
composite: [ImageMagick-7.1.1-36](#) -- [description](#)
config_data: [Module::Build-0.4234](#) -- [description](#)
conjure: [ImageMagick-7.1.1-36](#) -- [description](#)
convert: [ImageMagick-7.1.1-36](#) -- [description](#)
convert-dtsv0: [dtc-1.7.1](#) -- [description](#)
copydatabase: [Xapi안-1.4.26](#) -- [description](#)
corepack: [Node.js-20.16.0](#) -- [description](#)
cpack: [CMake-3.30.2](#) -- [description](#)
cpio: [cpio-2.15](#) -- [description](#)
cracklib-check: [CrackLib-2.10.2](#) -- [description](#)
cracklib-format: [CrackLib-2.10.2](#) -- [description](#)

cracklib-packer:
crc32:
create-cracklib-dict:
createdb:
createuser:
cryptsetup:
cryptsetup-reencrypt:
cspctl:
ctags:
ctest:
cue2toc:
cups-browsed:
cups-calibrate:
cups-config:
cups-genppd.5.2:
cups-genppdupdate:
cupsaccept:
cupscctl:
cupsd:
cupsdisable:
cupsenable:
cupsfilter:
cupsreject:
cupstestppd:
curl:
curl-config:
cvlc:
cvt:
cvtsudoers:
cwebp:
cx18-ctl:
cxpm:
cygdb:
cython:
cythonize:
danetool:
dash:
db2*:
db4-entities.pl:
dbmmanage:
dbus-binding-tool:
dbwrap_tool:
dconf:
dconf-editor:
dcraw_emu:
dcraw_half:
ddns-confgen:
decode_tm6000:
delv:
depdiagram-generate:
depdiagram-generate-all:
depdiagram-prepare:
derb:
designer:
designer:
desktop-file-edit:
desktop-file-install:
desktop-file-validate:
desktoptojson:
devdump:
dhpcd:
diagtool:
diffimg:
diffpp:
dig:
dijkstra:
dirmngr:
dirmngr-client:
disable-paste:
display:
display-buttons:
display-coords:
jpeg:
djxl:
dl10k1:
dtest:

CrackLib-2.10.2 -- [description](#)
Archive::Zip-1.68 -- [description](#)
CrackLib-2.10.2 -- [description](#)
PostgreSQL-16.4 -- [description](#)
PostgreSQL-16.4 -- [description](#)
cryptsetup-2.7.4 -- [description](#)
cryptsetup-2.7.4 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
Emacs-29.4 -- [description](#)
CMake-3.30.2 -- [description](#)
Cdrdao-1.2.4 -- [description](#)
cups-browsed-2.0.1 -- [description](#)
Gutenprint-5.3.4 -- [description](#)
Cups-2.4.10 -- [description](#)
Gutenprint-5.3.4 -- [description](#)
Gutenprint-5.3.4 -- [description](#)
Cups-2.4.10 -- [description](#)
curl-8.9.1 -- [description](#)
curl-8.9.1 -- [description](#)
VLC-3.0.21 -- [description](#)
libxcvt-0.1.2 -- [description](#)
Sudo-1.9.15p5 -- [description](#)
libwebp-1.4.0 -- [description](#)
v4l-utils-1.28.1 -- [description](#)
Xorg_Libraries -- [description](#)
Cython-3.0.11 -- [description](#)
Cython-3.0.11 -- [description](#)
Cython-3.0.11 -- [description](#)
GnuTLS-3.8.7.1 -- [description](#)
Dash-0.5.12 -- [description](#)
DocBook-utils-0.6.14 -- [description](#)
docbook-xml-5.1 -- [description](#)
Apache-2.4.62 -- [description](#)
dbus-glib-0.112 -- [description](#)
Samba-4.20.4 -- [description](#)
DConf-0.40.0 / DConf-Editor-45.0.1 -- [description](#)
DConf-0.40.0 / DConf-Editor-45.0.1 -- [description](#)
libraw-0.21.2 -- [description](#)
libraw-0.21.2 -- [description](#)
BIND-9.20.0 -- [description](#)
v4l-utils-1.28.1 -- [description](#)
BIND-9.20.0 -- [description](#)
Building KDE Frameworks 6.5.0 (KF6) -- [description](#)
Building KDE Frameworks 6.5.0 (KF6) -- [description](#)
Building KDE Frameworks 6.5.0 (KF6) -- [description](#)
icu-75.1 -- [description](#)
qt5-components-5.15.14 -- [description](#)
Qt-6.7.2 -- [description](#)
desktop-file-utils-0.27 -- [description](#)
desktop-file-utils-0.27 -- [description](#)
desktop-file-utils-0.27 -- [description](#)
Building KDE Frameworks 6.5.0 (KF6) -- [description](#)
Cdrtools-3.02a09 -- [description](#)
dhpcd-10.0.8 -- [description](#)
LLVM-18.1.7 -- [description](#)
Graphviz-12.1.0 -- [description](#)
Enscript-1.6.6 -- [description](#)
BIND-9.20.0 -- [description](#)
Graphviz-12.1.0 -- [description](#)
GnuPG-2.4.5 -- [description](#)
GnuPG-2.4.5 -- [description](#)
GPM-1.20.7 -- [description](#)
ImageMagick-7.1.1-36 -- [description](#)
GPM-1.20.7 -- [description](#)
GPM-1.20.7 -- [description](#)
libjpeg-turbo-3.0.1 -- [description](#)
libjxl-0.10.3 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
unixODBC-2.3.12 -- [description](#)

dmeventd: [LVM2-2.03.26](#) -- [description](#)
dmsetup: [LVM2-2.03.26](#) -- [description](#)
dnssec-cds: [BIND-9.20.0](#) -- [description](#)
dnssec-dsfromkey: [BIND-9.20.0](#) -- [description](#)
dnssec-importkey: [BIND-9.20.0](#) -- [description](#)
dnssec-keyfromlabel: [BIND-9.20.0](#) -- [description](#)
dnssec-keymgr: [BIND-9.20.0](#) -- [description](#)
dnssec-ksr: [BIND-9.20.0](#) -- [description](#)
dnssec-revoke: [BIND-9.20.0](#) -- [description](#)
dnssec-settime: [BIND-9.20.0](#) -- [description](#)
dnssec-signzone: [BIND-9.20.0](#) -- [description](#)
dnssec-verify: [DocBook-utils-0.6.14](#) -- [description](#)
docbook2*: [docutils-0.21.2](#) -- [description](#)
docutils: [dolphin-24.08.0](#) -- [description](#)
dolphin: [dos2unix-7.5.2](#) -- [description](#)
dos2unix: [Graphviz-12.1.0](#) -- [description](#)
dot: [Graphviz-12.1.0](#) -- [description](#)
dot2gxl: [Dovecot-2.3.21.1](#) -- [description](#)
doveadm: [Dovecot-2.3.21.1](#) -- [description](#)
doveconf: [Dovecot-2.3.21.1](#) -- [description](#)
dovecot: [Dovecot-2.3.21.1](#) -- [description](#)
dovecot-sysreport: [Doxygen-1.12.0](#) -- [description](#)
doxygen: [Doxygen-1.12.0](#) -- [description](#)
doxyindexer: [Doxygen-1.12.0](#) -- [description](#)
doxypypp: [Doxygen-1.12.0](#) -- [description](#)
doxyqml: [Doxygen-1.12.0](#) -- [description](#)
doxysearch.cgi: [Doxygen-1.12.0](#) -- [description](#)
doxywizard: [Doxygen-1.12.0](#) -- [description](#)
drill: [Idns-1.8.4](#) -- [description](#)
driverless: [cups-filters-2.0.1](#) -- [description](#)
driverless-fax: [cups-filters-2.0.1](#) -- [description](#)
dropdb: [PostgreSQL-16.4](#) -- [description](#)
dropuser: [PostgreSQL-16.4](#) -- [description](#)
dsymutil: [LLVM-18.1.7](#) -- [description](#)
dsync: [Dovecot-2.3.21.1](#) -- [description](#)
dtc: [dtc-1.7.1](#) -- [description](#)
dtdiff: [dtc-1.7.1](#) -- [description](#)
dubdv: [Libdv-1.0.0](#) -- [description](#)
dumpmscat: [Samba-4.20.4](#) -- [description](#)
dumpnames: [Text::BibTeX-0.89](#) -- [description](#)
dumpsexp: [libgcrypt-1.11.0](#) -- [description](#)
dvb-fe-tool: [v4l-utils-1.28.1](#) -- [description](#)
dvb-format-convert: [v4l-utils-1.28.1](#) -- [description](#)
dvbv5-scan: [v4l-utils-1.28.1](#) -- [description](#)
dvbv5-zap: [v4l-utils-1.28.1](#) -- [description](#)
dvconnect: [Libdv-1.0.0](#) -- [description](#)
dvisvgm: [dvisvgm-3.4](#) -- [description](#)
dwebp: [libweebp-1.4.0](#) -- [description](#)
ebrowse: [Emacs-29.4](#) -- [description](#)
ecpg: [PostgreSQL-16.4](#) -- [description](#)
ed: [Ed-1.20.2](#) -- [description](#)
edgepaint: [Graphviz-12.1.0](#) -- [description](#)
editcap: [Wireshark-4.2.6](#) -- [description](#)
editmap: [sendmail-8.18.1](#) -- [description](#)
efibootdump: [Contents](#)
efibootmgr: [efibootmgr-18](#) -- [description](#)
efisecdb: [efivar-39](#) -- [description](#)
efivar: [efivar-39](#) -- [description](#)
elf2dmp: [gemu-9.0.2](#) -- [description](#)
emacs: [Emacs-29.4](#) -- [description](#)
emacsclient: [Emacs-29.4](#) -- [description](#)
enchant-2: [Emacs-29.4](#) -- [description](#)
enchant-lsmod-2: [enchant-2.8.2](#) -- [description](#)
encodedev: [enchant-2.8.2](#) -- [description](#)
enscript: [Libdv-1.0.0](#) -- [description](#)
eog: [Enscript-1.6.6](#) -- [description](#)
epdfview: [EOG-45.4](#) -- [description](#)
epiphany: [ePDFView-gtk3-20200814](#) -- [description](#)
erb: [Epiphany-46.3](#) -- [description](#)
escapesrc: [Ruby-3.3.4](#) -- [description](#)
escputil: [icu-75.1](#) -- [description](#)
etags: [Gutenprint-5.3.4](#) -- [description](#)
eventlogadm: [Emacs-29.4](#) -- [description](#)
evince: [Samba-4.20.4](#) -- [description](#)
evince-previewer: [Evince-46.3.1](#) -- [description](#)
evince-thumbnailer: [Evince-46.3.1](#) -- [description](#)

evolution: Evolution-3.52.4 -- [description](#)
exempi: Exempi-2.6.5 -- [description](#)
exicyclog: Exim-4.98 -- [description](#)
exigrep: Exim-4.98 -- [description](#)
exim: Exim-4.98 -- [description](#)
exim-4.98-2: Exim-4.98 -- [description](#)
eximon: Exim-4.98 -- [description](#)
eximon.bin: Exim-4.98 -- [description](#)
eximstats: Exim-4.98 -- [description](#)
exim_checkaccess: Exim-4.98 -- [description](#)
exim_dbmbuild: Exim-4.98 -- [description](#)
exim_dumpdb: Exim-4.98 -- [description](#)
exim_fixdb: Exim-4.98 -- [description](#)
exim_lock: Exim-4.98 -- [description](#)
exim_tidydb: Exim-4.98 -- [description](#)
exinext: Exim-4.98 -- [description](#)
exipick: Exim-4.98 -- [description](#)
exiqgrep: Exim-4.98 -- [description](#)
exiqsumm: Exim-4.98 -- [description](#)
exiv2: Exiv2-0.28.3 -- [description](#)
exiwhat: Exim-4.98 -- [description](#)
exo-desktop-item-edit: Exo-4.18.0 -- [description](#)
exo-open: Exo-4.18.0 -- [description](#)
exportfs: NFS-Utils-2.6.4 -- [description](#)
extract_a52: Liba52-0.8.0 -- [description](#)
extract_mpeg2: libmpeg2-0.5.1 -- [description](#)
f2py: NumPy-2.1.0 -- [description](#)
faac: FAAC-1_30 -- [description](#)
faad: FAAD2-2.11.1 -- [description](#)
faillock: Linux-PAM-1.6.1 -- [description](#)
falkon: falkon-24.08.0 -- [description](#)
fancontrol: Lm-sensors-3-6-0 -- [description](#)
fatlabel: dosfstools-4.2 -- [description](#)
fbrun: Fluxbox-1.3.7 -- [description](#)
fbsetbg: Fluxbox-1.3.7 -- [description](#)
fbsetroot: Fluxbox-1.3.7 -- [description](#)
fbxine: xine-ui-0.99.14 -- [description](#)
fc-cache: Fontconfig-2.15.0 -- [description](#)
fc-cat: Fontconfig-2.15.0 -- [description](#)
fc-conflict: Fontconfig-2.15.0 -- [description](#)
fc-list: Fontconfig-2.15.0 -- [description](#)
fc-match: Fontconfig-2.15.0 -- [description](#)
fc-pattern: Fontconfig-2.15.0 -- [description](#)
fc-query: Fontconfig-2.15.0 -- [description](#)
fc-scan: Fontconfig-2.15.0 -- [description](#)
fc-validate: Fontconfig-2.15.0 -- [description](#)
fcgistarterm: Apache-2.4.62 -- [description](#)
fcron: Fcron-3.2.1 -- [description](#)
fcrondyn: Fcron-3.2.1 -- [description](#)
fcronsighup: Fcron-3.2.1 -- [description](#)
fcrontab: Fcron-3.2.1 -- [description](#)
fdp: Graphviz-12.1.0 -- [description](#)
fdtdump: dtc-1.7.1 -- [description](#)
fdtget: dtc-1.7.1 -- [description](#)
fdtoverlay: dtc-1.7.1 -- [description](#)
fdtput: dtc-1.7.1 -- [description](#)
feh: feh-3.10.3 -- [description](#)
fetchmail: Fetchmail-6.4.39 -- [description](#)
fetchmailconf: Fetchmail-6.4.39 -- [description](#)
ffmpeg: FFmpeg-7.0.2 -- [description](#)
ffplay: FFmpeg-7.0.2 -- [description](#)
ffprobe: FFmpeg-7.0.2 -- [description](#)
fftw-wisdom: fftw-3.3.10 -- [description](#)
fftw-wisdom-to-conf: fftw-3.3.10 -- [description](#)
file-roller: File-Roller-44.3 -- [description](#)
FileCheck: LLVM-18.1.7 -- [description](#)
findrule: File::Find::Rule-0.34 -- [description](#)
firefox: Firefox-128.1.0esr -- [description](#)
fix-qdf: Qpdf-11.9.1 -- [description](#)
fixparts: gptfdisk-1.0.10 -- [description](#)
fixqt4headers.pl: qt5-components-5.15.14 -- [description](#)
flac: FLAC-1.4.3 -- [description](#)
flea: Mutt-2.2.13 -- [description](#)
fitk-config: FLTK-1.3.9 -- [description](#)
fluid: FLTK-1.3.9 -- [description](#)
fluxbox: Fluxbox-1.3.7 -- [description](#)

fluxbox-generate_menu: Fluxbox-1.3.7 -- [description](#)
fluxbox-remote: Fluxbox-1.3.7 -- [description](#)
fluxbox-update_configs: Fluxbox-1.3.7 -- [description](#)
fontforge: FontForge-20230101 -- [description](#)
fontimage: FontForge-20230101 -- [description](#)
fontlint: FontForge-20230101 -- [description](#)
foomatic-rip: cups-filters-2.0.1 -- [description](#)
fop: fop-2.9 -- [description](#)
formail: Procmail-3.24 -- [description](#)
freerdp-proxy: FreeRDP-3.7.0 -- [description](#)
freerdp-shadow-cli: FreeRDP-3.7.0 -- [description](#)
freetype2-config: FreeType-2.13.3 -- [description](#)
fribidi: FriBidi-1.0.15 -- [description](#)
fsadm: LVM2-2.03.26 -- [description](#)
fsck.btrfs: btrfs-progs-6.10.1 -- [description](#)
fsck.jfs: jfsutils-1.1.15 -- [description](#)
fsck.xfs: xfsprogs-6.9.0 -- [description](#)
fsck.fat: dosfstools-4.2 -- [description](#)
fsidd: NFS-Utils-2.6.4 -- [description](#)
ftpasswd: ProFTPD-1.3.8b -- [description](#)
ftpcount: ProFTPD-1.3.8b -- [description](#)
ftpdctl: ProFTPD-1.3.8b -- [description](#)
ftpmail: ProFTPD-1.3.8b -- [description](#)
ftpquota: ProFTPD-1.3.8b -- [description](#)
ftpscrub: ProFTPD-1.3.8b -- [description](#)
ftpshut: ProFTPD-1.3.8b -- [description](#)
ftptop: ProFTPD-1.3.8b -- [description](#)
ftpwho: ProFTPD-1.3.8b -- [description](#)
funzip: UnZip-6.0 -- [description](#)
fusermount3: Fuse-3.16.2 -- [description](#)
g-ir-annotation-tool: GLib-2.80.4 -- [description](#)
g-ir-compiler: GLib-2.80.4 -- [description](#)
g-ir-doc-tool: GLib-2.80.4 -- [description](#)
g-ir-generate: GLib-2.80.4 -- [description](#)
g-ir-inspect: GLib-2.80.4 -- [description](#)
g-ir-scanner: GLib-2.80.4 -- [description](#)
g13: GnuPG-2.4.5 -- [description](#)
galera_new_cluster: MariaDB-10.11.8 -- [description](#)
galera_recovery: MariaDB-10.11.8 -- [description](#)
gamma4scanimage: SANE-1.2.1 -- [description](#)
application: GLib-2.80.4 -- [description](#)
gc: Graphviz-12.1.0 -- [description](#)
gcalccmd: gnome-calculator-46.1 -- [description](#)
gccgo: GCC-14.2.0 -- [description](#)
gcm-import: gnome-color-manager-3.36.0 -- [description](#)
gcm-inspect: gnome-color-manager-3.36.0 -- [description](#)
gcm-picker: gnome-color-manager-3.36.0 -- [description](#)
gcm-viewer: gnome-color-manager-3.36.0 -- [description](#)
gcore: GDB-15.1 -- [description](#)
gcr-viewer: Gcr-3.41.2 -- [description](#)
gcr-viewer-gtk4: Gcr-4.3.0 -- [description](#)
gdb-add-index: GDB-15.1 -- [description](#)
gdb-prog: GDB-15.1 -- [description](#)
gdbserver: GDB-15.1 -- [description](#)
gdbus: GLib-2.80.4 -- [description](#)
gdbus-codegen: GLib-2.80.4 -- [description](#)
gdisk: gptfdisk-1.0.10 -- [description](#)
gdk-pixbuf-csource: gdk-pixbuf-2.42.12 -- [description](#)
gdk-pixbuf-pixdata: gdk-pixbuf-2.42.12 -- [description](#)
gdk-pixbuf-query-loaders: gdk-pixbuf-2.42.12 -- [description](#)
gdk-pixbuf-thumbnailer: gdk-pixbuf-2.42.12 -- [description](#)
gdm: GDM-46.2 -- [description](#)
gdm-control: openbox-3.6.1 -- [description](#)
gdm-screenshot: GDM-46.2 -- [description](#)
gedit: Gedit-47.0 -- [description](#)
gegl: geogl-0.4.48 -- [description](#)
gegl-imgcmp: geogl-0.4.48 -- [description](#)
gem: Ruby-3.3.4 -- [description](#)
genbrk: icu-75.1 -- [description](#)
genccode: icu-75.1 -- [description](#)
gencfu: icu-75.1 -- [description](#)
gencmn: icu-75.1 -- [description](#)
gencnval: icu-75.1 -- [description](#)
gendict: icu-75.1 -- [description](#)
genl-ctrl-list: libnl-3.10.0 -- [description](#)
gennorm2: icu-75.1 -- [description](#)

genrb:
gensprep:
gentest:
get-versions:
getcifsacl:
gfortran:
gi-compile-repository:
gi-decompile-typelib:
gi-docgen:
gi-inspect-typelib:
gif2rgb:
gif2webp:
gifbuild:
gifclrmp:
giffix:
giftext:
giftool:
gimp-2.99:
gimp-console-2.99:
gimptool-2.99:
gio:
gio-querymodules:
git:
git-clang-format:
git-cvsserver:
git-receive-pack:
git-shell:
git-upload-archive:
git-upload-pack:
gitk:
gjs-console:
glad:
glewinfo:
glib-compile-resources:
glib-compile-schemas:
glib-genmarshal:
glib-gettextize:
glib-mkenums:
glslang:
glxgears:
glxinfo:
gm2:
gml2gv:
gnome-calculator:
gnome-connections:
gnome-control-center:
gnome-disk-image-mounter:
gnome-disks:
gnome-keyring-daemon:
gnome-logs:
gnome-maps:
gnome-nettool:
gnome-panel-control:
gnome-power-statistics:
gnome-screenshot:
gnome-session:
gnome-session-inhibit:
gnome-session-quit:
gnome-shell:
gnome-system-monitor:
gnome-terminal:
gnome-tweaks:
gnome-weather:
gnumeric:
gnumeric-1.12.57:
gnutls-cli:
gnutls-cli-debug:
gnutls-serv:
go:
goa-daemon:
gobject-query:
gofmt:
gparted:
gparted_polkit:
gpg:
gpg-agent:

[icu-75.1](#) -- [description](#)
[icu-75.1](#) -- [description](#)
[Samba-4.20.4](#) -- [description](#)
[GPM-1.20.7](#) -- [description](#)
[cifs-utils-7.0](#) -- [description](#)
[GCC-14.2.0](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[Gi-DocGen-2024.1](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[glib-5.2.2](#) -- [description](#)
[libwebp-1.4.0](#) -- [description](#)
[glib-5.2.2](#) -- [description](#)
[Gimp-20240711](#) -- [description](#)
[Gimp-20240711](#) -- [description](#)
[Gimp-20240711](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[Git-2.46.0](#) -- [description](#)
[LLVM-18.1.7](#) -- [description](#)
[Git-2.46.0](#) -- [description](#)
[Gjs-1.80.2](#) -- [description](#)
[Glad-2.0.6](#) -- [description](#)
[GLEW-2.2.0](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[Glslang-14.3.0](#) -- [description](#)
[Mesa-24.1.5](#) -- [description](#)
[Mesa-24.1.5](#) -- [description](#)
[GCC-14.2.0](#) -- [description](#)
[Graphviz-12.1.0](#) -- [description](#)
[gnome-calculator-46.1](#) -- [description](#)
[gnome-connections-46.0](#) -- [description](#)
[gnome-control-center-46.4](#) -- [description](#)
[gnome-disk-utility-46.0](#) -- [description](#)
[gnome-disk-utility-46.0](#) -- [description](#)
[gnome-keyring-46.2](#) -- [description](#)
[gnome-logs-45.0](#) -- [description](#)
[gnome-maps-46.11](#) -- [description](#)
[gnome-nettool-42.0](#) -- [description](#)
[openbox-3.6.1](#) -- [description](#)
[gnome-power-manager-43.0](#) -- [description](#)
[gnome-screenshot-41.0](#) -- [description](#)
[gnome-session-46.0](#) -- [description](#)
[gnome-session-46.0](#) -- [description](#)
[gnome-session-46.0](#) -- [description](#)
[gnome-shell-46.4](#) -- [description](#)
[gnome-system-monitor-46.0](#) -- [description](#)
[gnome-terminal-3.52.2](#) -- [description](#)
[gnome-tweaks-46.1](#) -- [description](#)
[gnome-weather-46.0](#) -- [description](#)
[Gnumeric-1.12.57](#) -- [description](#)
[Gnumeric-1.12.57](#) -- [description](#)
[GnuTLS-3.8.7.1](#) -- [description](#)
[GnuTLS-3.8.7.1](#) -- [description](#)
[GnuTLS-3.8.7.1](#) -- [description](#)
[GCC-14.2.0](#) -- [description](#)
[gnome-online-accounts-3.50.4](#) -- [description](#)
[GLib-2.80.4](#) -- [description](#)
[GCC-14.2.0](#) -- [description](#)
[Gparted-1.6.0](#) -- [description](#)
[Gparted-1.6.0](#) -- [description](#)
[GnuPG-2.4.5](#) -- [description](#)
[GnuPG-2.4.5](#) -- [description](#)

gpg-card: GnuPG-2.4.5 -- [description](#)
gpg-connect-agent: GnuPG-2.4.5 -- [description](#)
gpg-error: libgpg-error-1.50 -- [description](#)
gpg-wks-client: GnuPG-2.4.5 -- [description](#)
gpg-wks-server: GnuPG-2.4.5 -- [description](#)
gpgconf: GnuPG-2.4.5 -- [description](#)
gpgme-json: GPGME-1.23.2 -- [description](#)
gpgme-tool: GPGME-1.23.2 -- [description](#)
gpgparsemail: GnuPG-2.4.5 -- [description](#)
gpgrt-config: libgpg-error-1.50 -- [description](#)
gpgscm: GnuPG-2.4.5 -- [description](#)
gpgsm: GnuPG-2.4.5 -- [description](#)
gpgsplit: GnuPG-2.4.5 -- [description](#)
gptar: GnuPG-2.4.5 -- [description](#)
gpgv: GnuPG-2.4.5 -- [description](#)
gpm: GPM-1.20.7 -- [description](#)
gpm-root: GPM-1.20.7 -- [description](#)
gr2fonttest: Graphite2-1.3.14 -- [description](#)
graphml2gv: Graphviz-12.1.0 -- [description](#)
gresource: GLib-2.80.4 -- [description](#)
grilo-test-ui: Grilo-0.3.16 -- [description](#)
grl-inspect: Grilo-0.3.16 -- [description](#)
grl-launch: Grilo-0.3.16 -- [description](#)
growisofs: dvd+rw-tools-7.1 -- [description](#)
grub-mkfont: GRUB-2.12 for EFI -- [description](#)
grub-mount: GRUB-2.12 for EFI -- [description](#)
gs: ghostscript-10.03.1 -- [description](#)
gsettings: GLib-2.80.4 -- [description](#)
gsf: libgsf-1.14.52 -- [description](#)
gsf-office-thumbnailer: libgsf-1.14.52 -- [description](#)
gsf-vba-dump: libgsf-1.14.52 -- [description](#)
gsl-config: gsl-2.8 -- [description](#)
gsl-histogram: gsl-2.8 -- [description](#)
gsl-randist: gsl-2.8 -- [description](#)
gsound-play: GSound-1.0.3 -- [description](#)
gspell-app1: gspell-1.12.2 -- [description](#)
gss-client: MIT Kerberos V5-1.21.3 -- [description](#)
gss-server: MIT Kerberos V5-1.21.3 -- [description](#)
gst-device-monitor-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-discoverer-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-inspect-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-launch-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-play-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-stats-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-tester-1.0: gst-plugins-base-1.24.7 -- [description](#)
gst-transcoder-1.0: gst-plugins-bad-1.24.7 -- [description](#)
gst-typefind-1.0: gstreamer-1.24.7 -- [description](#)
gtester: GLib-2.80.4 -- [description](#)
gtester-report: GLib-2.80.4 -- [description](#)
gtf: Xorg-Server-21.1.13 -- [description](#)
gtk-builder-tool: GTK+-3.24.43 -- [description](#)
gtk-encode-symbolic-svg: GTK+-3.24.43 -- [description](#)
gtk-launch: GTK+-3.24.43 -- [description](#)
gtk-query-immodules-3.0: GTK+-3.24.43 -- [description](#)
gtk-query-settings: GTK+-3.24.43 -- [description](#)
gtk-update-icon-cache: GTK+-3.24.43 -- [description](#)
gtk3-demo: GTK+-3.24.43 -- [description](#)
gtk3-demo-application: GTK+-3.24.43 -- [description](#)
gtk3-icon-browser: GTK+-3.24.43 -- [description](#)
gtk3-widget-factory: GTK+-3.24.43 -- [description](#)
gtk4-broadwayd: GTK-4.14.5 -- [description](#)
gtk4-builder-tool: GTK-4.14.5 -- [description](#)
gtk4-demo: GTK-4.14.5 -- [description](#)
gtk4-demo-application: GTK-4.14.5 -- [description](#)
gtk4-encode-symbolic-svg: GTK-4.14.5 -- [description](#)
gtk4-icon-browser: GTK-4.14.5 -- [description](#)
gtk4-launch: GTK-4.14.5 -- [description](#)
gtk4-node-editor: GTK-4.14.5 -- [description](#)
gtk4-print-editor: GTK-4.14.5 -- [description](#)
gtk4-query-settings: GTK-4.14.5 -- [description](#)
gtk4-update-icon-cache: GTK-4.14.5 -- [description](#)
gtk4-widget-factory: GTK-4.14.5 -- [description](#)
gtkdoc*: GTK-Doc-1.34.0 -- [description](#)
gucharmap: Gucharmap-15.1.5 -- [description](#)
guild: Guile-3.0.10 -- [description](#)
guile: Guile-3.0.10 -- [description](#)

guile-config:
guile-snarf:
guile-tools:
gusbcmd:
gv2gml:
gv2gxl:
gvcolor:
gvedit:
gvgen:
gview:
gvim:
gvimdiff:
gvimtutor:
gvmap:
gvmap.sh:
gvnccapture:
gvpack:
gvpr:
gwenview:
gwenview_importer:
gx12dot:
gx12gv:
half_mt:
handy-1-demo:
hatchling:
hb-info:
hb-ot-shape-closure:
hb-shape:
hb-subset:
hb-view:
hda-verb:
hdajackretask:
hdparm:
hdspconf:
hdsploader:
hdspmixer:
hex2hcd:
hg:
highlight:
highlight-gui:
httest:
hmac256:
hmaptool:
homectl:
host:
hoststat:
htcacheclean:
htdbm:
htdigest:
htpasswd:
httpd:
htttx2dbm:
hwmixvolume:
ibus-daemon:
ibus-setup:
iceauth:
icehelp:
icesh:
icesound:
icewm:
icewm-menu-fdo:
icewm-menu-xrandr:
icewm-session:
icewm-set-gnomewm:
icewmgb:
icewmhint:
icon-name-mapping:
icu-config:
icuinfo:
icupkg:
id3convert:
id3cp:
id3info:
id3tag:
identify:
idl2wrs:

Guile-3.0.10 -- [description](#)
Guile-3.0.10 -- [description](#)
Guile-3.0.10 -- [description](#)
libgusb-0.4.9 -- [description](#)
Graphviz-12.1.0 -- [description](#)
Vim-9.1.0660 -- [description](#)
Vim-9.1.0660 -- [description](#)
Vim-9.1.0660 -- [description](#)
Vim-9.1.0660 -- [description](#)
Graphviz-12.1.0 -- [description](#)
Graphviz-12.1.0 -- [description](#)
gtk-vnc-1.3.1 -- [description](#)
Graphviz-12.1.0 -- [description](#)
Graphviz-12.1.0 -- [description](#)
Graphviz-12.1.0 -- [description](#)
gwenview-24.08.0 -- [description](#)
gwenview-24.08.0 -- [description](#)
Graphviz-12.1.0 -- [description](#)
Graphviz-12.1.0 -- [description](#)
libraw-0.21.2 -- [description](#)
libhandy-1.8.3 -- [description](#)
Hatchling-1.25.0 -- [description](#)
harfBuzz-9.0.0 -- [description](#)
harfBuzz-9.0.0 -- [description](#)
harfBuzz-9.0.0 -- [description](#)
harfBuzz-9.0.0 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
Hdparm-9.65 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
BlueZ-5.77 -- [description](#)
Mercurial-6.8.1 -- [description](#)
Highlight-4.13 -- [description](#)
Highlight-4.13 -- [description](#)
GPM-1.20.7 -- [description](#)
libgcrypt-1.11.0 -- [description](#)
LLVM-18.1.7 -- [description](#)
Systemd-256.4 -- [description](#)
BIND-9.20.0 -- [description](#)
sendmail-8.18.1 -- [description](#)
Apache-2.4.62 -- [description](#)
alsa-tools-1.2.11 -- [description](#)
ibus-1.5.30 -- [description](#)
ibus-1.5.30 -- [description](#)
Xorg_Applications -- [description](#)
IceWM-3.6.0 -- [description](#)
Icon-naming-utils-0.8.90 -- [description](#)
icu-75.1 -- [description](#)
icu-75.1 -- [description](#)
icu-75.1 -- [description](#)
id3lib-3.8.3 -- [description](#)
id3lib-3.8.3 -- [description](#)
id3lib-3.8.3 -- [description](#)
id3lib-3.8.3 -- [description](#)
ImageMagick-7.1.1-36 -- [description](#)
Wireshark-4.2.6 -- [description](#)

idle3: Python-3.12.5 -- [description](#)
idn: libidn-1.42 -- [description](#)
idn2: libidn2-2.3.7 -- [description](#)
iecset: alsu-utils-1.2.12 -- [description](#)
ifrename: Wireless Tools-29 -- [description](#)
img2webp: libwebp-1.4.0 -- [description](#)
imgcmp: JasPer-4.2.4 -- [description](#)
imginfo: JasPer-4.2.4 -- [description](#)
imlib2_bumpmap: imlib2-1.12.3 -- [description](#)
imlib2_colorspace: imlib2-1.12.3 -- [description](#)
imlib2_conv: imlib2-1.12.3 -- [description](#)
imlib2_grab: imlib2-1.12.3 -- [description](#)
imlib2_load: imlib2-1.12.3 -- [description](#)
imlib2_poly: imlib2-1.12.3 -- [description](#)
imlib2_show: imlib2-1.12.3 -- [description](#)
imlib2_test: imlib2-1.12.3 -- [description](#)
imlib2_view: imlib2-1.12.3 -- [description](#)
import: ImageMagick-7.1.1-36 -- [description](#)
initdb: PostgreSQL-16.4 -- [description](#)
init_audigy*: alsu-tools-1.2.11 -- [description](#)
init_live: alsu-tools-1.2.11 -- [description](#)
inkscape: Inkscape-1.3.2 -- [description](#)
inkview: Inkscape-1.3.2 -- [description](#)
innochecksum: MariaDB-10.11.8 -- [description](#)
install-catalog: sgml-common-0.6.3 -- [description](#)
integritysetup: cryptsetup-2.7.4 -- [description](#)
intercept-build: LLVM-18.1.7 -- [description](#)
iostat: Sysstat-12.7.6 -- [description](#)
ip6tables: iptables-1.8.10 -- [description](#)
ipmaddr: Net-tools-2.10 -- [description](#)
ippeveprinter: Cups-2.4.10 -- [description](#)
ippfind: Cups-2.4.10 -- [description](#)
iptool: Cups-2.4.10 -- [description](#)
iptables: iptables-1.8.10 -- [description](#)
iptables-apply: iptables-1.8.10 -- [description](#)
iptables-legacy: iptables-1.8.10 -- [description](#)
iptables-legacy-restore: iptables-1.8.10 -- [description](#)
iptables-legacy-save: iptables-1.8.10 -- [description](#)
iptables-restore: iptables-1.8.10 -- [description](#)
iptables-save: iptables-1.8.10 -- [description](#)
iptables-xml: iptables-1.8.10 -- [description](#)
iptunnel: Net-tools-2.10 -- [description](#)
ir-keytable: y4l-utils-1.28.1 -- [description](#)
irb: Ruby-3.3.4 -- [description](#)
isadump: Lm-sensors-3-6-0 -- [description](#)
isaset: Lm-sensors-3-6-0 -- [description](#)
iso-info: libcdio-2.1.0 -- [description](#)
iso-read: libcdio-2.1.0 -- [description](#)
isodebug: Cdrtools-3.02a09 -- [description](#)
isodump: Cdrtools-3.02a09 -- [description](#)
isoinfo: Cdrtools-3.02a09 -- [description](#)
isovfy: Cdrtools-3.02a09 -- [description](#)
ispell: Aspell-0.60.8.1 -- [description](#)
isql: unixODBC-2.3.12 -- [description](#)
itstool: itstool-2.0.7 -- [description](#)
iusql: unixODBC-2.3.12 -- [description](#)
ivtv-ctl: y4l-utils-1.28.1 -- [description](#)
iw: iw-6.9 -- [description](#)
iwconfig: Wireless Tools-29 -- [description](#)
iwevent: Wireless Tools-29 -- [description](#)
iwgetid: Wireless Tools-29 -- [description](#)
iwlist: Wireless Tools-29 -- [description](#)
iwpriv: Wireless Tools-29 -- [description](#)
iwspy: Wireless Tools-29 -- [description](#)
jade: OpenJade-1.3.2 -- [description](#)
jar: OpenJDK-22.0.2 -- [description](#)
jarsigner: OpenJDK-22.0.2 -- [description](#)
jasper: JasPer-4.2.4 -- [description](#)
java: OpenJDK-22.0.2 -- [description](#)
javac: OpenJDK-22.0.2 -- [description](#)
javadoc: OpenJDK-22.0.2 -- [description](#)
javap: OpenJDK-22.0.2 -- [description](#)
jcmd: OpenJDK-22.0.2 -- [description](#)
jconsole: OpenJDK-22.0.2 -- [description](#)
jdb: OpenJDK-22.0.2 -- [description](#)
jdeprscan: OpenJDK-22.0.2 -- [description](#)

jdeps: OpenJDK-22.0.2 -- description
jfr: OpenJDK-22.0.2 -- description
jfs_debugfs: jfsutils-1.1.15 -- description
jfs_fsck: jfsutils-1.1.15 -- description
jfs_fscklog: jfsutils-1.1.15 -- description
jfs_logdump: jfsutils-1.1.15 -- description
jfs_mkfs: jfsutils-1.1.15 -- description
jfs_tune: jfsutils-1.1.15 -- description
jhsdb: OpenJDK-22.0.2 -- description
jimage: OpenJDK-22.0.2 -- description
jinfo: JasPer-4.2.4 -- description
jiv: OpenJDK-22.0.2 -- description
jlink: JOE-4.6 -- description
jmacs: OpenJDK-22.0.2 -- description
jmap: OpenJDK-22.0.2 -- description
jmod: OpenJDK-22.0.2 -- description
joe: JOE-4.6 -- description
jpackage: OpenJDK-22.0.2 -- description
jpegtran: libjpeg-turbo-3.0.1 -- description
jpgicc: Little CMS-2.16 -- description
jpico: JOE-4.6 -- description
jps: OpenJDK-22.0.2 -- description
jrundscript: OpenJDK-22.0.2 -- description
js115 : SpiderMonkey from firefox-115.14.0 -- description
js115-config : SpiderMonkey from firefox-115.14.0 -- description
jshell: OpenJDK-22.0.2 -- description
json-glib-format: JSON-GLib-1.8.0 -- description
json-glib-validate: JSON-GLib-1.8.0 -- description
jstack: OpenJDK-22.0.2 -- description
jstar: JOE-4.6 -- description
jstat: OpenJDK-22.0.2 -- description
jstatd: OpenJDK-22.0.2 -- description
jw: DocBook-utils-0.6.14 -- description
jwebserver: OpenJDK-22.0.2 -- description
jxlinfo: libjxl-0.10.3 -- description
k3b: k3b-24.08.0 -- description
k5srvutil: MIT Kerberos V5-1.21.3 -- description
kacpimon: acpid-2.0.34 -- description
kadmin: MIT Kerberos V5-1.21.3 -- description
kadmin.local: MIT Kerberos V5-1.21.3 -- description
kadmind: MIT Kerberos V5-1.21.3 -- description
kate: kate-24.08.0 -- description
kbuildsycoca6: Building KDE Frameworks 6.5.0 (KF6). -- description
kbxutil: GnuPG-2.4.5 -- description
kcookiejar6: Building KDE Frameworks 6.5.0 (KF6). -- description
kdb5_ldap_util: MIT Kerberos V5-1.21.3 -- description
kdb5_util: MIT Kerberos V5-1.21.3 -- description
kded6: Building KDE Frameworks 6.5.0 (KF6). -- description
kdeinit6: Building KDE Frameworks 6.5.0 (KF6). -- description
kdenlive: kdenlive-24.08.0 -- description
kdenlive_render: kdenlive-24.08.0 -- description
kdestroy: MIT Kerberos V5-1.21.3 -- description
kea-admin: Kea 2.6.1 DHCP Server -- description
kea-ctrl-agent: Kea 2.6.1 DHCP Server -- description
kea-dhcp-ddns: Kea 2.6.1 DHCP Server -- description
kea-dhcp4: Kea 2.6.1 DHCP Server -- description
kea-dhcp6: Kea 2.6.1 DHCP Server -- description
kea-lfc: Kea 2.6.1 DHCP Server -- description
keactrl: Kea 2.6.1 DHCP Server -- description
keashell: Kea 2.6.1 DHCP Server -- description
key.dns_resolver: keyutils-1.6.3 -- description
keyctl: keyutils-1.6.3 -- description
keytool: OpenJDK-22.0.2 -- description
kf6-config: Building KDE Frameworks 6.5.0 (KF6). -- description
kf6kross: Building KDE Frameworks 6.5.0 (KF6). -- description
kgendesignerplugin: Building KDE Frameworks 6.5.0 (KF6). -- description
kglobalaccel6: Building KDE Frameworks 6.5.0 (KF6). -- description
khelpcenter: khelphcenter-24.08.0 -- description
kinit: MIT Kerberos V5-1.21.3 -- description
kjs6: Building KDE Frameworks 6.5.0 (KF6). -- description
kjscmd6: Building KDE Frameworks 6.5.0 (KF6). -- description
kjsconsole: Building KDE Frameworks 6.5.0 (KF6). -- description
klist: MIT Kerberos V5-1.21.3 -- description
koi8rxterm: xterm-393 -- description
konsole: Konsole-24.08.0 -- description
konsoleprofile: Konsole-24.08.0 -- description

konversation: konversation-24.08.0 -- [description](#)
kpackagelauncherqml: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
kpackageTool6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
kpasswd: MIT_Kerberos_V5-1.21.3 -- [description](#)
kprop: MIT_Kerberos_V5-1.21.3 -- [description](#)
kpropd: MIT_Kerberos_V5-1.21.3 -- [description](#)
kproplog: MIT_Kerberos_V5-1.21.3 -- [description](#)
krb5-send-pr: MIT_Kerberos_V5-1.21.3 -- [description](#)
krb5-config: MIT_Kerberos_V5-1.21.3 -- [description](#)
krb5kdc: MIT_Kerberos_V5-1.21.3 -- [description](#)
kreadconfig6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
kscreen-doctor: kconfig-6.5.0_for_lxqt -- [description](#)
kshell6: libkscreen-6.1.4_for_lxqt -- [description](#)
ksu: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
kswitch: MIT_Kerberos_V5-1.21.3 -- [description](#)
ktelnetservice6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
ktrash6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
ktutil: MIT_Kerberos_V5-1.21.3 -- [description](#)
kvno: MIT_Kerberos_V5-1.21.3 -- [description](#)
kwalletd6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
kwrite: kate-24.08.0 -- [description](#)
kwriteconfig6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
kconfig-6.5.0 for lxqt -- [description](#)
I2ping: BlueZ-5.77 -- [description](#)
I2test: BlueZ-5.77 -- [description](#)
I4p-temp1: Log::Log4perl-1.57 -- [description](#)
lame: LAME-3.100 -- [description](#)
lconvert: qt5-components-5.15.14 -- [description](#)
lconvert: Qt-6.7.2 -- [description](#)
ld10k1: alsa-tools-1.2.11 -- [description](#)
ld10k1d: alsa-tools-1.2.11 -- [description](#)
ldapadd: OpenLDAP-2.6.8 -- [description](#)
ldapcompare: OpenLDAP-2.6.8 -- [description](#)
ldapdelete: OpenLDAP-2.6.8 -- [description](#)
ldapexop: OpenLDAP-2.6.8 -- [description](#)
ldapmodify: OpenLDAP-2.6.8 -- [description](#)
ldapmodrdn: OpenLDAP-2.6.8 -- [description](#)
ldappasswd: OpenLDAP-2.6.8 -- [description](#)
ldapsearch: OpenLDAP-2.6.8 -- [description](#)
ldapurl: OpenLDAP-2.6.8 -- [description](#)
ldapvc: OpenLDAP-2.6.8 -- [description](#)
ldapwhoami: OpenLDAP-2.6.8 -- [description](#)
ldbadd: Samba-4.20.4 -- [description](#)
ldbdel: Samba-4.20.4 -- [description](#)
ldbedit: Samba-4.20.4 -- [description](#)
ldbmodify: Samba-4.20.4 -- [description](#)
ldbrename: Samba-4.20.4 -- [description](#)
ldbsearch: Samba-4.20.4 -- [description](#)
ldns-config: Idns-1.8.4 -- [description](#)
libevdev-tweak-device: libevdev 1.13.2 -- [description](#)
libinput: libinput-1.26.1 -- [description](#)
libndp: libndp-1.9 -- [description](#)
libnewt: newt-0.52.24 -- [description](#)
libpng-config: libpng-1.6.43 -- [description](#)
libreoffice-suite: LibreOffice-24.8.0 -- [description](#)
librest-demo: rest-0.9.1 -- [description](#)
libwacom-list-devices: libwacom-2.12.2 -- [description](#)
libwacom-list-local-devices: libwacom-2.12.2 -- [description](#)
libwacom-show-stylus: libwacom-2.12.2 -- [description](#)
libwacom-update-db: libwacom-2.12.2 -- [description](#)
lightdm: lightdm-1.32.0 -- [description](#)
lightdm-gtk-greeter: lightdm-1.32.0 -- [description](#)
linguist: qt5-components-5.15.14 -- [description](#)
linguist: Qt-6.7.2 -- [description](#)
linkicc: Little_CMS-2.16 -- [description](#)
links: Links-2.30 -- [description](#)
llc: LLVM-18.1.7 -- [description](#)
lli: LLVM-18.1.7 -- [description](#)
llvm-addr2line: LLVM-18.1.7 -- [description](#)
llvm-ar: LLVM-18.1.7 -- [description](#)
llvm-as: LLVM-18.1.7 -- [description](#)
llvm-bcanalyzer: LLVM-18.1.7 -- [description](#)
llvm-bitcode-strip: LLVM-18.1.7 -- [description](#)
llvm-c-test: LLVM-18.1.7 -- [description](#)
llvm-cat: LLVM-18.1.7 -- [description](#)

[llvm-cfi-verify](#): LLVM-18.1.7 -- [description](#)
[llvm-config](#): LLVM-18.1.7 -- [description](#)
[llvm-cov](#): LLVM-18.1.7 -- [description](#)
[llvm-cvtres](#): LLVM-18.1.7 -- [description](#)
[llvm-cxxdump](#): LLVM-18.1.7 -- [description](#)
[llvm-cxxfilt](#): LLVM-18.1.7 -- [description](#)
[llvm-cxxmap](#): LLVM-18.1.7 -- [description](#)
[llvm-debuginfo-analyzer](#): LLVM-18.1.7 -- [description](#)
[llvm-debuginfod](#): LLVM-18.1.7 -- [description](#)
[llvm-debuginfod-find](#): LLVM-18.1.7 -- [description](#)
[llvm-diff](#): LLVM-18.1.7 -- [description](#)
[llvm-dis](#): LLVM-18.1.7 -- [description](#)
[llvm-dwarfdump](#): LLVM-18.1.7 -- [description](#)
[llvm-dwarfutil](#): LLVM-18.1.7 -- [description](#)
[llvm-dwp](#): LLVM-18.1.7 -- [description](#)
[llvm-elfabi](#): LLVM-18.1.7 -- [description](#)
[llvm-exegesis](#): LLVM-18.1.7 -- [description](#)
[llvm-extract](#): LLVM-18.1.7 -- [description](#)
[llvm-gsymutil](#): LLVM-18.1.7 -- [description](#)
[llvm-ifs](#): LLVM-18.1.7 -- [description](#)
[llvm-install-name-tool](#): LLVM-18.1.7 -- [description](#)
[llvm-jitlink](#): LLVM-18.1.7 -- [description](#)
[llvm-libtool-darwin](#): LLVM-18.1.7 -- [description](#)
[llvm-link](#): LLVM-18.1.7 -- [description](#)
[llvm-lipo](#): LLVM-18.1.7 -- [description](#)
[llvm-lto](#): LLVM-18.1.7 -- [description](#)
[llvm-lto2](#): LLVM-18.1.7 -- [description](#)
[llvm-mc](#): LLVM-18.1.7 -- [description](#)
[llvm-mca](#): LLVM-18.1.7 -- [description](#)
[llvm-ml](#): LLVM-18.1.7 -- [description](#)
[llvm-modextract](#): LLVM-18.1.7 -- [description](#)
[llvm-mt](#): LLVM-18.1.7 -- [description](#)
[llvm-nm](#): LLVM-18.1.7 -- [description](#)
[llvm-objcopy](#): LLVM-18.1.7 -- [description](#)
[llvm-objdump](#): LLVM-18.1.7 -- [description](#)
[llvm-opt-report](#): LLVM-18.1.7 -- [description](#)
[llvm-pdbutil](#): LLVM-18.1.7 -- [description](#)
[llvm-propdata](#): LLVM-18.1.7 -- [description](#)
[llvm-profgen](#): LLVM-18.1.7 -- [description](#)
[llvm-ranlib](#): LLVM-18.1.7 -- [description](#)
[llvm-rc](#): LLVM-18.1.7 -- [description](#)
[llvm-readobj](#): LLVM-18.1.7 -- [description](#)
[llvm-readtapi](#): LLVM-18.1.7 -- [description](#)
[llvm-reduce](#): LLVM-18.1.7 -- [description](#)
[llvm-remarkutil](#): LLVM-18.1.7 -- [description](#)
[llvm-rtdyld](#): LLVM-18.1.7 -- [description](#)
[llvm-size](#): LLVM-18.1.7 -- [description](#)
[llvm-spirv](#): SPIRV-LLVM-Translator-18.1.3 -- [description](#)
[llvm-split](#): LLVM-18.1.7 -- [description](#)
[llvm-stress](#): LLVM-18.1.7 -- [description](#)
[llvm-strings](#): LLVM-18.1.7 -- [description](#)
[llvm-symbolizer](#): LLVM-18.1.7 -- [description](#)
[llvm-tblgen](#): LLVM-18.1.7 -- [description](#)
[llvm-tli-checker](#): LLVM-18.1.7 -- [description](#)
[llvm-undname](#): LLVM-18.1.7 -- [description](#)
[llvm-xray](#): LLVM-18.1.7 -- [description](#)
[lo10k1](#): als-tools-1.2.11 -- [description](#)
[lobase](#): LibreOffice-24.8.0 -- [description](#)
[localc](#): LibreOffice-24.8.0 -- [description](#)
[lockfile](#): Procmail-3.24 -- [description](#)
[locktest](#): Samba-4.20.4 -- [description](#)
[lodraw](#): LibreOffice-24.8.0 -- [description](#)
[logresolve](#): Apache-2.4.62 -- [description](#)
[logrotate](#): Logrotate-3.22.0 -- [description](#)
[loimpress](#): LibreOffice-24.8.0 -- [description](#)
[lomath](#): LibreOffice-24.8.0 -- [description](#)
[loweb](#): LibreOffice-24.8.0 -- [description](#)
[lowntfs-3g](#): ntfs-3g-2022.10.3 -- [description](#)
[lowriter](#): LibreOffice-24.8.0 -- [description](#)
[lp](#): Cups-2.4.10 -- [description](#)
[lpadmin](#): Cups-2.4.10 -- [description](#)
[lpcl](#): Cups-2.4.10 -- [description](#)
[lpinfo](#): Cups-2.4.10 -- [description](#)
[lpmove](#): Cups-2.4.10 -- [description](#)
[lpoptions](#): Cups-2.4.10 -- [description](#)
[lpq](#): Cups-2.4.10 -- [description](#)

lpr: [Cups-2.4.10](#) -- [description](#)
lprm: [Cups-2.4.10](#) -- [description](#)
lpstat: [Cups-2.4.10](#) -- [description](#)
lrelease-pro: [qt5-components-5.15.14](#) -- [description](#)
lrelease: [qt5-components-5.15.14](#) -- [description](#)
lrelease: [Qt-6.7.2](#) -- [description](#)
lsb_release: [LSB-Tools-0.12](#) -- [description](#)
lsof: [lsof-4.99.0](#) -- [description](#)
lspci: [pciutils-3.13.0](#) -- [description](#)
lsusb: [usbutils-017](#) -- [description](#)
lsusb.py: [usbutils-017](#) -- [description](#)
lua: [Lua-5.4.7](#) -- [description](#)
lua5.2: [Lua-5.2.4](#) -- [description](#)
luac: [Lua-5.4.7](#) -- [description](#)
luac5.2: [Lua-5.2.4](#) -- [description](#)
lupdate-pro: [qt5-components-5.15.14](#) -- [description](#)
lupdate: [qt5-components-5.15.14](#) -- [description](#)
lupdate: [Qt-6.7.2](#) -- [description](#)
lvm: [LVM2-2.03.26](#) -- [description](#)
lvm-cache-stats: [libblockdev-3.1.1](#) -- [description](#)
lvmdump: [LVM2-2.03.26](#) -- [description](#)
lwp-download: [LWP \(libwww-perl-6.77\)](#) -- [description](#)
lwp-dump: [LWP \(libwww-perl-6.77\)](#) -- [description](#)
lwp-mirror: [LWP \(libwww-perl-6.77\)](#) -- [description](#)
lwp-request: [LWP \(libwww-perl-6.77\)](#) -- [description](#)
lxdoas: [lxqt-sudo-2.0.0](#) -- [description](#)
lximage-qt: [lximage-qt-2.0.1](#) -- [description](#)
lxqt-about: [lxqt-about-2.0.0](#) -- [description](#)
lxqt-admin-time: [lxqt-admin-2.0.0](#) -- [description](#)
lxqt-admin-user: [lxqt-admin-2.0.0](#) -- [description](#)
lxqt-archiver: [lxqt-archiver-1.0.0](#) -- [description](#)
lxqt-backlight_backend: [liblxqt-2.0.0](#) -- [description](#)
lxqt-config: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-appearance: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-brightness: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-file-associations: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-globalkeyshortcuts: [lxqt-globalkeys-2.0.0](#) -- [description](#)
lxqt-config-input: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-locale: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-monitor: [lxqt-config-2.0.0](#) -- [description](#)
lxqt-config-notificationd: [lxqt-notificationd-2.0.1](#) -- [description](#)
lxqt-config-powermanagement: [lxqt-powermanagement-2.0.0](#) -- [description](#)
lxqt-config-session: [lxqt-session-2.0.0](#) -- [description](#)
lxqt-globalkeysd: [lxqt-globalkeys-2.0.0](#) -- [description](#)
lxqt-leave: [lxqt-session-2.0.0](#) -- [description](#)
lxqt-notificationd: [lxqt-notificationd-2.0.1](#) -- [description](#)
lxqt.openssh-askpass: [lxqt-openssh-askpass-2.0.1](#) -- [description](#)
lxqt-panel: [lxqt-panel-2.0.1](#) -- [description](#)
lxqt-policykit-agent: [lxqt-policykit-2.0.0](#) -- [description](#)
lxqt-powermanagement: [lxqt-powermanagement-2.0.0](#) -- [description](#)
lxqt-runner: [lxqt-runner-2.0.0](#) -- [description](#)
lxqt-session: [lxqt-session-2.0.0](#) -- [description](#)
lxqt-sudo: [lxqt-sudo-2.0.0](#) -- [description](#)
lxqt2-transupdate: [lxqt-build-tools-2.0.0](#) -- [description](#)
lxsu: [lxqt-sudo-2.0.0](#) -- [description](#)
lxsudo: [lxqt-sudo-2.0.0](#) -- [description](#)
lynx: [Lynx-2.9.2](#) -- [description](#)
mac2unix: [dos2unix-7.5.2](#) -- [description](#)
magick: [ImageMagick-7.1.1-36](#) -- [description](#)
Magick-config: [ImageMagick-7.1.1-36](#) -- [description](#)
mail: [mailx-12.5](#) -- [description](#)
mailq: [Postfix-3.9.0](#) -- [description](#)
mailstat: [sendmail-8.18.1](#) -- [description](#)
mailstats: [Procmail-3.24](#) -- [description](#)
mailx: [sendmail-8.18.1](#) -- [description](#)
make-ca: [mailx-12.5](#) -- [description](#)
makeconv: [make-ca-1.14](#) -- [description](#)
makemap: [icu-75.1](#) -- [description](#)
mako-render: [sendmail-8.18.1](#) -- [description](#)
mandoc: [Mako-1.3.5](#) -- [description](#)
mariabackup: [mandoc-1.14.6](#) -- [description](#)
mariadb-access: [MariaDB-10.11.8](#) -- [description](#)
mariadb-admin: [MariaDB-10.11.8](#) -- [description](#)
mariadb-binlog: [MariaDB-10.11.8](#) -- [description](#)
mariadb-check: [MariaDB-10.11.8](#) -- [description](#)

mariadb-client-test-embedded: MariaDB-10.11.8 -- [description](#)
mariadb-client-test: MariaDB-10.11.8 -- [description](#)
mariadb-conv: MariaDB-10.11.8 -- [description](#)
mariadb-convert-table-format: MariaDB-10.11.8 -- [description](#)
mariadb-dump: MariaDB-10.11.8 -- [description](#)
mariadb-dumpsslow: MariaDB-10.11.8 -- [description](#)
mariadb-find-rows: MariaDB-10.11.8 -- [description](#)
mariadb-fix-extensions: MariaDB-10.11.8 -- [description](#)
mariadb-hotcopy: MariaDB-10.11.8 -- [description](#)
mariadb-import: MariaDB-10.11.8 -- [description](#)
mariadb-install-db: MariaDB-10.11.8 -- [description](#)
mariadb-ldb: MariaDB-10.11.8 -- [description](#)
mariadb: MariaDB-10.11.8 -- [description](#)
mariadb-plugin: MariaDB-10.11.8 -- [description](#)
mariadb-secure-installation: MariaDB-10.11.8 -- [description](#)
mariadb-service-convert: MariaDB-10.11.8 -- [description](#)
mariadb-setpermission: MariaDB-10.11.8 -- [description](#)
mariadb-show: MariaDB-10.11.8 -- [description](#)
mariadb-slap: MariaDB-10.11.8 -- [description](#)
mariadb-test-embedded: MariaDB-10.11.8 -- [description](#)
mariadb-test: MariaDB-10.11.8 -- [description](#)
mariadb-tzinfo-to-sql: MariaDB-10.11.8 -- [description](#)
mariadb-upgrade: MariaDB-10.11.8 -- [description](#)
mariadb-waitpid: MariaDB-10.11.8 -- [description](#)
mariabdb-embedded: MariaDB-10.11.8 -- [description](#)
mariabd: MariaDB-10.11.8 -- [description](#)
mariabdb-multi: MariaDB-10.11.8 -- [description](#)
mariabdb-safe: MariaDB-10.11.8 -- [description](#)
mariadb_config: MariaDB-10.11.8 -- [description](#)
markdown_py: Markdown-3.6 -- [description](#)
masktest: Samba-4.20.4 -- [description](#)
mbim-network: libmbim-1.26.4 -- [description](#)
mbimcli: libmbim-1.26.4 -- [description](#)
mbstream: MariaDB-10.11.8 -- [description](#)
mc: MC-4.8.32 -- [description](#)
mcdiff: MC-4.8.32 -- [description](#)
mcedit: MC-4.8.32 -- [description](#)
mcview: MC-4.8.32 -- [description](#)
mdadm: mdadm-4.3 -- [description](#)
mdig: BIND-9.20.0 -- [description](#)
mdmon: mdadm-4.3 -- [description](#)
mdsearch: Samba-4.20.4 -- [description](#)
media-ctl: v4l-utils-1.28.1 -- [description](#)
meinproc6: Building KDE Frameworks 6.5.0 (KF6) -- [description](#)
melt: mlt-7.26.0 -- [description](#)
mem_image: libraw-0.21.2 -- [description](#)
mergecap: Wireshark-4.2.6 -- [description](#)
meshdebug: Qt-6.7.2 -- [description](#)
metaflac: FLAC-1.4.3 -- [description](#)
mev: GPM-1.20.7 -- [description](#)
mii-tool: Net-tools-2.10 -- [description](#)
mixartloader: alsa-tools-1.2.11 -- [description](#)
mkafmmap: Enscript-1.6.6 -- [description](#)
mkbimap: Potrace-1.16 -- [description](#)
mkfontdir: Xorg_Applications -- [description](#)
mkfontscale: Xorg_Applications -- [description](#)
mkfs.btrfs: btrfs-progs-6.10.1 -- [description](#)
mkfs.jfs: jfsutils-1.1.15 -- [description](#)
mkfs.ntfs: ntfs-3g-2022.10.3 -- [description](#)
mkfs.xfs: xfsprogs-6.9.0 -- [description](#)
mkfs.fat: dosfstools-4.2 -- [description](#)
mkhomedir_helper: Linux-PAM-1.6.1 -- [description](#)
mkhybrid: Cdrtools-3.02a09 -- [description](#)
mkisofs: Cdrtools-3.02a09 -- [description](#)
mkntfs: ntfs-3g-2022.10.3 -- [description](#)
mkpasswd: Whois-5.4.3 -- [description](#)
mm2gv: Graphviz-12.1.0 -- [description](#)
mmc-tool: libcdio-2.1.0 -- [description](#)
mmcli: ModemManager-1.18.12 -- [description](#)
moc: qt5-components-5.15.14 -- [description](#)
ModemManager: ModemManager-1.18.12 -- [description](#)
mogrify: ImageMagick-7.1.1-36 -- [description](#)
montage: ImageMagick-7.1.1-36 -- [description](#)
mount.cifs: cifs-utils-7.0 -- [description](#)
mount.fuse3: Fuse-3.16.2 -- [description](#)
mount.lowntfs-3g: ntfs-3g-2022.10.3 -- [description](#)

mount.nfs: NFS-Utils-2.6.4 -- [description](#)
mount.nfs4: NFS-Utils-2.6.4 -- [description](#)
mount.ntfs: ntfs-3g-2022.10.3 -- [description](#)
mount.ntfs-3g: ntfs-3g-2022.10.3 -- [description](#)
mount.smb3: cifs-utils-7.0 -- [description](#)
mountstats: NFS-Utils-2.6.4 -- [description](#)
mouse-dpi-tool: libevdev-1.13.2 -- [description](#)
mouse-test: GPM-1.20.7 -- [description](#)
mousepad: Mousepad-0.6.2 -- [description](#)
mozcerts-qt6: Qca-2.3.9 -- [description](#)
mp3rtp: LAME-3.100 -- [description](#)
mpeg2dec: libmpeg2-0.5.1 -- [description](#)
mpg123: mpg123-1.32.7 -- [description](#)
mpg123-id3dump: mpg123-1.32.7 -- [description](#)
mpg123-strip: mpg123-1.32.7 -- [description](#)
out123: mpg123-1.32.7 -- [description](#)
mpicalc: libgcrypt-1.11.0 -- [description](#)
mpstat: Sysstat-12.7.6 -- [description](#)
mpv: mpv-0.38.0 -- [description](#)
msql2mysql: MariaDB-10.11.8 -- [description](#)
ms_print: Valgrind-3.23.0 -- [description](#)
mt: cpio-2.15 -- [description](#)
mtdev-test: mtdev-1.1.7 -- [description](#)
multirender_test: libraw-0.21.2 -- [description](#)
mupdf: MuPDF-1.24.8 -- [description](#)
mupdf-gl: MuPDF-1.24.8 -- [description](#)
mupdf-x11: MuPDF-1.24.8 -- [description](#)
muraster: MuPDF-1.24.8 -- [description](#)
mutool: MuPDF-1.24.8 -- [description](#)
mutt: Mutt-2.2.13 -- [description](#)
muttbug: Mutt-2.2.13 -- [description](#)
mutter: Mutter-46.4 -- [description](#)
mutt_dotlock: Mutt-2.2.13 -- [description](#)
mutt_pgpring: Mutt-2.2.13 -- [description](#)
mvxattr: Samba-4.20.4 -- [description](#)
myisamchk: MariaDB-10.11.8 -- [description](#)
myisamlog: MariaDB-10.11.8 -- [description](#)
myisampack: MariaDB-10.11.8 -- [description](#)
myisam_ftdump: MariaDB-10.11.8 -- [description](#)
mysql_config: MariaDB-10.11.8 -- [description](#)
mytop: MariaDB-10.11.8 -- [description](#)
my_print_defaults: MariaDB-10.11.8 -- [description](#)
nail: mailx-12.5 -- [description](#)
named: BIND-9.20.0 -- [description](#)
named-checkconf: BIND-9.20.0 -- [description](#)
named-checkzone: BIND-9.20.0 -- [description](#)
named-compilezone: BIND-9.20.0 -- [description](#)
named-journalprint: BIND-9.20.0 -- [description](#)
named-nzd2nzf: BIND-9.20.0 -- [description](#)
named-rrchecker: BIND-9.20.0 -- [description](#)
nameif: Net-tools-2.10 -- [description](#)
nano: Nano-8.1 -- [description](#)
nasm: NASM-2.16.03 -- [description](#)
nautilus: Nautilus-46.2 -- [description](#)
ncat: Nmap-7.95 -- [description](#)
ncftp: NcFTP-3.2.7 -- [description](#)
ncftpbatch: NcFTP-3.2.7 -- [description](#)
ncftpbookmarks: NcFTP-3.2.7 -- [description](#)
ncftpget: NcFTP-3.2.7 -- [description](#)
ncftpls: NcFTP-3.2.7 -- [description](#)
ncftpput: NcFTP-3.2.7 -- [description](#)
ncftpspooler: NcFTP-3.2.7 -- [description](#)
ndiff: Nmap-7.95 -- [description](#)
ndisasm: NASM-2.16.03 -- [description](#)
ndptool: libndp-1.9 -- [description](#)
ndrdump: Samba-4.20.4 -- [description](#)
neato: Graphviz-12.1.0 -- [description](#)
neon-config: neon-0.33.0 -- [description](#)
net: Samba-4.20.4 -- [description](#)
netstat: Net-tools-2.10 -- [description](#)
nettle-hash: Nettle-3.10 -- [description](#)
nettle-ifib-stream: Nettle-3.10 -- [description](#)
nettle-pbkdf2: Nettle-3.10 -- [description](#)
NetworkManager: NetworkManager-1.48.8 -- [description](#)
newaliases: Postfix-3.9.0 -- [description](#)
sendmail: sendmail-8.18.1 -- [description](#)

nfsconf: NFS-Utils-2.6.4 -- [description](#)
nfsdclnts: NFS-Utils-2.6.4 -- [description](#)
nfsiostat: NFS-Utils-2.6.4 -- [description](#)
nfsstat: NFS-Utils-2.6.4 -- [description](#)
nfsynproxy: iptables-1.8.10 -- [description](#)
nl-class-add: libnl-3.10.0 -- [description](#)
nl-class-delete: libnl-3.10.0 -- [description](#)
nl-class-list: libnl-3.10.0 -- [description](#)
nl-classid-lookup: libnl-3.10.0 -- [description](#)
nl-cls-add: libnl-3.10.0 -- [description](#)
nl-cls-delete: libnl-3.10.0 -- [description](#)
nl-cls-list: libnl-3.10.0 -- [description](#)
nl-link-list: libnl-3.10.0 -- [description](#)
nl-pktloc-lookup: libnl-3.10.0 -- [description](#)
nl-qdisc-add: libnl-3.10.0 -- [description](#)
nl-qdisc-delete: libnl-3.10.0 -- [description](#)
nl-qdisc-list: libnl-3.10.0 -- [description](#)
nm-connection-editor: network-manager-applet-1.34.0 -- [description](#)
nm-online: NetworkManager-1.48.8 -- [description](#)
nmap: Nmap-7.95 -- [description](#)
nmbd: Samba-4.20.4 -- [description](#)
nmblookup: Samba-4.20.4 -- [description](#)
nmcli: NetworkManager-1.48.8 -- [description](#)
nmtui: NetworkManager-1.48.8 -- [description](#)
nmtui-connect: NetworkManager-1.48.8 -- [description](#)
nmtui-edit: NetworkManager-1.48.8 -- [description](#)
nmtui-hostname: NetworkManager-1.48.8 -- [description](#)
node: Node.js-20.16.0 -- [description](#)
nop: Graphviz-12.1.0 -- [description](#)
normalizer: Charset-normalizer-3.3.2 -- [description](#)
notify-send: libnotify-0.8.3 -- [description](#)
nping: Nmap-7.95 -- [description](#)
npm: Node.js-20.16.0 -- [description](#)
npth-config: npth-1.7 -- [description](#)
nsec3hash: BIND-9.20.0 -- [description](#)
nsgmls: OpenSP-1.5.2 -- [description](#)
nslookup: BIND-9.20.0 -- [description](#)
nspr-config: NSPR-4.35 -- [description](#)
nss-config: NSS-3.103 -- [description](#)
nsupdate: BIND-9.20.0 -- [description](#)
ntfs-3g: ntfs-3g-2022.10.3 -- [description](#)
ntfs-3g.probe: ntfs-3g-2022.10.3 -- [description](#)
ntfscat: ntfs-3g-2022.10.3 -- [description](#)
ntfsclone: ntfs-3g-2022.10.3 -- [description](#)
ntfscluster: ntfs-3g-2022.10.3 -- [description](#)
ntfscmp: ntfs-3g-2022.10.3 -- [description](#)
ntfscp: ntfs-3g-2022.10.3 -- [description](#)
ntfsfix: ntfs-3g-2022.10.3 -- [description](#)
ntfsinfo: ntfs-3g-2022.10.3 -- [description](#)
ntfslabel: ntfs-3g-2022.10.3 -- [description](#)
ntfsls: ntfs-3g-2022.10.3 -- [description](#)
ntfsresize: ntfs-3g-2022.10.3 -- [description](#)
ntfsundelete: ntfs-3g-2022.10.3 -- [description](#)
ntlm_auth: Samba-4.20.4 -- [description](#)
ntp-keygen: ntp-4.2.8p18 -- [description](#)
ntp-wait: ntp-4.2.8p18 -- [description](#)
ntpd: ntp-4.2.8p18 -- [description](#)
ntpdate: ntp-4.2.8p18 -- [description](#)
ntpdc: ntp-4.2.8p18 -- [description](#)
ntpq: ntp-4.2.8p18 -- [description](#)
ntptime: ntp-4.2.8p18 -- [description](#)
ntptrace: ntp-4.2.8p18 -- [description](#)
nvlc: VLC-3.0.21 -- [description](#)
nvptx-arch: LLVM-18.1.7 -- [description](#)
obconf-qt: obconf-qt-0.16.4 -- [description](#)
obxprop: openbox-3.6.1 -- [description](#)
ocsptool: GnuTLS-3.8.7.1 -- [description](#)
odbcinst: unixODBC-2.3.12 -- [description](#)
odbc_config: unixODBC-2.3.12 -- [description](#)
ogg123: vorbis-tools-1.4.2 -- [description](#)
oggdec: vorbis-tools-1.4.2 -- [description](#)
oggenc: vorbis-tools-1.4.2 -- [description](#)
ogginfo: vorbis-tools-1.4.2 -- [description](#)
oid2name: PostgreSQL-16.4 -- [description](#)
okular: okular-24.08.0 -- [description](#)
oLschema2ldif: Samba-4.20.4 -- [description](#)

onsgmls: OpenSP-1.5.2 -- description
on_ac_power: pm-utils-1.4.1 -- description
openbox: openbox-3.6.1 -- description
openbox-gnome-session: openbox-3.6.1 -- description
openbox-kde-session: openbox-3.6.1 -- description
openbox-session: openbox-3.6.1 -- description
openjade: OpenJade-1.3.2 -- description
opj_compress: OpenJPEG-2.5.2 -- description
opj_decompress: OpenJPEG-2.5.2 -- description
opj_dump: OpenJPEG-2.5.2 -- description
opt: LLVM-18.1.7 -- description
osage: Graphviz-12.1.0 -- description
osgmlnorm: OpenSP-1.5.2 -- description
osirrox: libisoburn-1.5.6 -- description
ospam: OpenSP-1.5.2 -- description
ospcat: OpenSP-1.5.2 -- description
ospent: OpenSP-1.5.2 -- description
osx: Enscript-1.6.6 -- description
over: p11-kit-0.25.5 -- description
p11-kit: GnuTLS-3.8.7.1 -- description
p11tool: PulseAudio-17.0 -- description
pa-info: PulseAudio-17.0 -- description
pacat: PulseAudio-17.0 -- description
pacmd: PulseAudio-17.0 -- description
pactl: PulseAudio-17.0 -- description
padsp: PulseAudio-17.0 -- description
pamon: PulseAudio-17.0 -- description
pam_namespace_helper: Linux-PAM-1.6.1 -- description
pam_timestamp_check: Linux-PAM-1.6.1 -- description
pango-list: Pango-1.54.0 -- description
pango-segmentation: Pango-1.54.0 -- description
pango-view: Pango-1.54.0 -- description
paper: libpaper-2.2.5 -- description
paperconf: libpaper-2.2.5 -- description
paplay: PulseAudio-17.0 -- description
paps: PulseAudio-17.0 -- description
parec: PulseAudio-17.0 -- description
parecord: PulseAudio-17.0 -- description
parole: PulseAudio-17.0 -- description
parted: Parole-4.18.1 -- description
partprobe: parted-3.6 -- description
pasuspender: parted-3.6 -- description
patchelf: PulseAudio-17.0 -- description
patchwork: Patchelf-0.18.0 -- description
pavucontrol: Graphviz-12.1.0 -- description
pavucontrol-qt: pavucontrol-6.1 -- description
pax: pavucontrol-qt-2.0.0 -- description
pax11publish: Pax-20240817 -- description
pcilmr: PulseAudio-17.0 -- description
pcmanfm-qt: pciutils-3.13.0 -- description
pcre2-config: pcmanfm-qt-2.0.0 -- description
pcre2grep: PCRE2-10.44 -- description
pcre2: PCRE2-10.44 -- description
pcxhrloader: PCRE2-10.44 -- description
pdbeedit: alsal-tools-1.2.11 -- description
pdfattach: Samba-4.20.4 -- description
pdfdetach: Poppler-24.08.0 -- description
pdffonts: Poppler-24.08.0 -- description
pdfimages: Poppler-24.08.0 -- description
pdfinfo: Poppler-24.08.0 -- description
pdftocairo: Poppler-24.08.0 -- description
pdftohtml: Poppler-24.08.0 -- description
pdftoppm: Poppler-24.08.0 -- description
pdftops: Poppler-24.08.0 -- description
pdftotext: Poppler-24.08.0 -- description
pdfunite: Poppler-24.08.0 -- description
pdfseparate: Poppler-24.08.0 -- description
pdfsig: PHP-8.3.10 -- description
pear: libpeas-1.36.0 -- description
peas-demo: MariaDB-10.11.8 -- description
perror: PostgreSQL-16.4 -- description
pgbench: Mutt-2.2.13 -- description
pgpwrap: PostgreSQL-16.4 -- description
pg_amcheck: PostgreSQL-16.4 -- description
pg_archivecleanup: PostgreSQL-16.4 -- description
pg_basebackup: PostgreSQL-16.4 -- description

pg_checksums: PostgreSQL-16.4 -- [description](#)
pg_config: PostgreSQL-16.4 -- [description](#)
pg_controldata: PostgreSQL-16.4 -- [description](#)
pg_ctl: PostgreSQL-16.4 -- [description](#)
pg_dump: PostgreSQL-16.4 -- [description](#)
pg_dumpall: PostgreSQL-16.4 -- [description](#)
pg_isready: PostgreSQL-16.4 -- [description](#)
pg_receivewal: PostgreSQL-16.4 -- [description](#)
pg_recvlogical: PostgreSQL-16.4 -- [description](#)
pg_resetwal: PostgreSQL-16.4 -- [description](#)
pg_restore: PostgreSQL-16.4 -- [description](#)
pg_rewind: PostgreSQL-16.4 -- [description](#)
pg_standby: PostgreSQL-16.4 -- [description](#)
pg_test_fsync: PostgreSQL-16.4 -- [description](#)
pg_test_timing: PostgreSQL-16.4 -- [description](#)
pg_upgrade: PostgreSQL-16.4 -- [description](#)
pg_verifybackup: PostgreSQL-16.4 -- [description](#)
pg_waldump: PostgreSQL-16.4 -- [description](#)
php: PHP-8.3.10 -- [description](#)
php-fpm: PHP-8.3.10 -- [description](#)
phpdbg: PHP-8.3.10 -- [description](#)
pidstat: Sysstat-12.7.6 -- [description](#)
pinentry: pinentry-1.3.1 -- [description](#)
pinentry-curses: pinentry-1.3.1 -- [description](#)
pinentry-emacs: pinentry-1.3.1 -- [description](#)
pinentry-fltk: pinentry-1.3.1 -- [description](#)
pinentry-gnome3: pinentry-1.3.1 -- [description](#)
pinentry-qt: pinentry-1.3.1 -- [description](#)
pinentry-tty: pinentry-1.3.1 -- [description](#)
pipewire: Pipewire-1.2.3 -- [description](#)
pipewire-pulse: Pipewire-1.2.3 -- [description](#)
pixeltool: qt5-components-5.15.14 -- [description](#)
pixeltool: Qt-6.7.2 -- [description](#)
pk12util: NSS-3.103 -- [description](#)
pkaction: Polkit-125 -- [description](#)
pkcheck: Polkit-125 -- [description](#)
pkcs1-conv: Nettle-3.10 -- [description](#)
pkexec: Polkit-125 -- [description](#)
pkgdata: icu-75.1 -- [description](#)
pktyagent: Polkit-125 -- [description](#)
plasma-activities: plasma-activities-6.1.4 -- [description](#)
plasmapkg2: Building KDE Frameworks 6.5.0 (KF6) -- [description](#)
playout: gst-plugins-bad-1.24.7 -- [description](#)
plipconfig: Net-tools-2.10 -- [description](#)
pltcl_delmod: PostgreSQL-16.4 -- [description](#)
pltcl_listmod: PostgreSQL-16.4 -- [description](#)
pltcl_loadmod: PostgreSQL-16.4 -- [description](#)
pluginviewer: Cyrus SASL-2.1.28 -- [description](#)
pm-hibernate: pm-utils-1.4.1 -- [description](#)
pm-is-supported: pm-utils-1.4.1 -- [description](#)
pm-powersave: pm-utils-1.4.1 -- [description](#)
pm-suspend: pm-utils-1.4.1 -- [description](#)
pm-suspend-hybrid: pm-utils-1.4.1 -- [description](#)
png-fix-itxt: libpng-1.6.43 -- [description](#)
pngfix: libpng-1.6.43 -- [description](#)
pnmixer: pnmixer-0.7.2 -- [description](#)
polkit-gnome-authentication-agent-1: polkit-gnome-0.105 -- [description](#)
polkitd: Polkit-125 -- [description](#)
postalias: Postfix-3.9.0 -- [description](#)
postcat: Postfix-3.9.0 -- [description](#)
postconf: Postfix-3.9.0 -- [description](#)
postdrop: Postfix-3.9.0 -- [description](#)
postfix: Postfix-3.9.0 -- [description](#)
postgres: PostgreSQL-16.4 -- [description](#)
postkick: Postfix-3.9.0 -- [description](#)
postlock: Postfix-3.9.0 -- [description](#)
postlog: Postfix-3.9.0 -- [description](#)
postmap: Postfix-3.9.0 -- [description](#)
postmulti: Postfix-3.9.0 -- [description](#)
postprocessing_benchmark: libraw-0.21.2 -- [description](#)
postqueue: Postfix-3.9.0 -- [description](#)
postsuper: Postfix-3.9.0 -- [description](#)
potrace: Potrace-1.16 -- [description](#)
powerprofilesctl: Contents
ppdc: Cups-2.4.10 -- [description](#)
pppd: libpppd-2.0.0 -- [description](#)

ppdhtml: Cups-2.4.10 -- [description](#)
ppdi: libppd-2.0.0 -- [description](#)
ppdmerge: Cups-2.4.10 -- [description](#)
ppdpo: libppd-2.0.0 -- [description](#)
praliases: Cups-2.4.10 -- [description](#)
precat: Aspell-0.60.8.1 -- [description](#)
preparetips6: Building_KDE_Frameworks_6.5.0_(KF6) -- [description](#)
preunzip: Aspell-0.60.8.1 -- [description](#)
prezip: Aspell-0.60.8.1 -- [description](#)
prezip-bin: Aspell-0.60.8.1 -- [description](#)
procmail: Procmail-3.24 -- [description](#)
profiles: Samba-4.20.4 -- [description](#)
proftpd: ProFTPD-1.3.8b -- [description](#)
protoc: Protobuf-27.3 -- [description](#)
protoc-c: Protobuf-c-1.5.0 -- [description](#)
prune: Graphviz-12.1.0 -- [description](#)
prxs: ProFTPD-1.3.8b -- [description](#)
psicc: Little_CMS-2.16 -- [description](#)
psktool: GnuTLS-3.8.7.1 -- [description](#)
psl: libpsl-0.21.5 -- [description](#)
pspell-config: Aspell-0.60.8.1 -- [description](#)
psql: PostgreSQL-16.4 -- [description](#)
pulseaudio: PulseAudio-17.0 -- [description](#)
purgestat: sendmail-8.18.1 -- [description](#)
pw-cat: Pipewire-1.2.3 -- [description](#)
pw-cli: Pipewire-1.2.3 -- [description](#)
pw-config: Pipewire-1.2.3 -- [description](#)
pw-dot: Pipewire-1.2.3 -- [description](#)
pw-dump: Pipewire-1.2.3 -- [description](#)
pw-jack: Pipewire-1.2.3 -- [description](#)
pw-link: Pipewire-1.2.3 -- [description](#)
pw-loopback: Pipewire-1.2.3 -- [description](#)
pw-metadata: Pipewire-1.2.3 -- [description](#)
pw-mididump: Pipewire-1.2.3 -- [description](#)
pw-mon: Pipewire-1.2.3 -- [description](#)
pw-profiler: Pipewire-1.2.3 -- [description](#)
pw-reserve: Pipewire-1.2.3 -- [description](#)
pw-top: Pipewire-1.2.3 -- [description](#)
pw-v4l2: Pipewire-1.2.3 -- [description](#)
pwhistory_helper: Linux-PAM-1.6.1 -- [description](#)
pwmake: libpwquality-1.4.5 -- [description](#)
pwmconfig: Lm-sensors-3-6-0 -- [description](#)
pwscore: libpwquality-1.4.5 -- [description](#)
pybabel: babel-2.16.0 -- [description](#)
pydo3c: Python-3.12.5 -- [description](#)
pygmentize: Pygments-2.18.0 -- [description](#)
pyserial-miniterm: pySerial-3.5 -- [description](#)
pyserial-ports: pySerial-3.5 -- [description](#)
pytest: Pytest-8.3.2 -- [description](#)
python3: Python-3.12.5 -- [description](#)
python3.11: Python-3.11.1 -- [description](#)
python3.12: Python-3.12.5 -- [description](#)
qalc: libqalculate-5.2.0 -- [description](#)
qcatoold-qt6: Qca-2.3.9 -- [description](#)
qcollectiongenerator-qt5: qt5-components-5.15.14 -- [description](#)
qcoro: qcoro-0.10.0
qdbus: qt5-components-5.15.14 -- [description](#)
qdbus: Qt-6.7.2 -- [description](#)
qdbuscpp2xml: qt5-components-5.15.14 -- [description](#)
qdbuscpp2xml: Qt-6.7.2 -- [description](#)
qdbusviewer: qt5-components-5.15.14 -- [description](#)
qdbusviewer: Qt-6.7.2 -- [description](#)
qdbusxml2cpp: qt5-components-5.15.14 -- [description](#)
qdbusxml2cpp: Qt-6.7.2 -- [description](#)
qdistanfieldgenerator: qt5-components-5.15.14 -- [description](#)
qdistanfieldgenerator: Qt-6.7.2 -- [description](#)
qdoc: qt5-components-5.15.14 -- [description](#)
qdoc: Qt-6.7.2 -- [description](#)
qemu-edid: qemu-9.0.2 -- [description](#)
qemu-ga: qemu-9.0.2 -- [description](#)
qemu-img: qemu-9.0.2 -- [description](#)
qemu-io: qemu-9.0.2 -- [description](#)

qemu-keymap: qemu-9.0.2 -- [description](#)
qemu-nbd: qemu-9.0.2 -- [description](#)
qemu-pr-helper: qemu-9.0.2 -- [description](#)
qemu-storage-daemon: qemu-9.0.2 -- [description](#)
qemu-system-x86_64: qemu-9.0.2 -- [description](#)
qhelpgenerator: qt5-components-5.15.14 -- [description](#)
qlalr: qt5-components-5.15.14 -- [description](#)
qmake: qt5-components-5.15.14 -- [description](#)
qmake: Qt-6.7.2 -- [description](#)
qmifirmware-update: libqmi-1.30.8 -- [description](#)
qmtnetwork: libqmi-1.30.8 -- [description](#)
qmcli: libqmi-1.30.8 -- [description](#)
qml: qt5-components-5.15.14 -- [description](#)
qml: Qt-6.7.2 -- [description](#)
qmlcachegen: qt5-components-5.15.14 -- [description](#)
qmleasing: qt5-components-5.15.14 -- [description](#)
qmleasing: Qt-6.7.2 -- [description](#)
qmlformat: qt5-components-5.15.14 -- [description](#)
qmlformat: Qt-6.7.2 -- [description](#)
qmlimportscanner: qt5-components-5.15.14 -- [description](#)
qmllint: qt5-components-5.15.14 -- [description](#)
qmllint: Qt-6.7.2 -- [description](#)
qmlmin: qt5-components-5.15.14 -- [description](#)
qmlplugindump: qt5-components-5.15.14 -- [description](#)
qmlplugindump: Qt-6.7.2 -- [description](#)
qmlpreview: qt5-components-5.15.14 -- [description](#)
qmlpreview: Qt-6.7.2 -- [description](#)
qmlprofiler: qt5-components-5.15.14 -- [description](#)
qmlprofiler: Qt-6.7.2 -- [description](#)
qmlscene: qt5-components-5.15.14 -- [description](#)
qmlscene: Qt-6.7.2 -- [description](#)
qmltestrunner: qt5-components-5.15.14 -- [description](#)
qmltestrunner: Qt-6.7.2 -- [description](#)
qpaeq: PulseAudio-17.0 -- [description](#)
qpdf: Qpdf-11.9.1 -- [description](#)
zlib-flate: Qpdf-11.9.1 -- [description](#)
qps: qps-2.9.0 -- [description](#)
qrencode: qrencode-4.1.1 -- [description](#)
qt-faststart: FFmpeg-7.0.2 -- [description](#)
qtattributionsscanner: qt5-components-5.15.14 -- [description](#)
qtdiag: qt5-components-5.15.14 -- [description](#)
qtdiag: Qt-6.7.2 -- [description](#)
qterminal: qterminal-2.0.1 -- [description](#)
qtpaths: qt5-components-5.15.14 -- [description](#)
qtpaths: Qt-6.7.2 -- [description](#)
qtplugininfo: qt5-components-5.15.14 -- [description](#)
qtplugininfo: Qt-6.7.2 -- [description](#)
qtwaylandscanner: qt5-components-5.15.14 -- [description](#)
QtWebEngineProcess: QtWebEngine-6.7.2 -- [description](#)
qtwebengine_convert_dict: QtWebEngine-6.7.2 -- [description](#)
qtxdg-mat: qtxdg-tools-4.0.0 -- [description](#)
quest: Xapian-1.4.26 -- [description](#)
qv4l2: v4l-utils-1.28.1 -- [description](#)
qvicap: v4l-utils-1.28.1 -- [description](#)
qvkgem: qt5-components-5.15.14 -- [description](#)
qvlc: VLC-3.0.21 -- [description](#)
rake: Ruby-3.3.4 -- [description](#)
randpkt: Wireshark-4.2.6 -- [description](#)
rapper: Raptor-2.0.16 -- [description](#)
rarp: Net-tools-2.10 -- [description](#)
rasql-config: Rasql-0.9.33 -- [description](#)
half_mt: libraw-0.21.2 -- [description](#)
rawshark: Wireshark-4.2.6 -- [description](#)
rcc: qt5-components-5.15.14 -- [description](#)
rctest: BlueZ-5.77 -- [description](#)
rdbg: Ruby-3.3.4 -- [description](#)
rdfproc: Redland-1.0.17 -- [description](#)
rdjpgcom: libjpeg-turbo-3.0.1 -- [description](#)
rdoc: Ruby-3.3.4 -- [description](#)
rds-ctl: v4l-utils-1.28.1 -- [description](#)
readcd: Cdrtools-3.02a09 -- [description](#)
red: Ed-1.20.2 -- [description](#)
redland-config: Redland-1.0.17 -- [description](#)
redland-db-upgrade: Redland-1.0.17 -- [description](#)
regdiff: Samba-4.20.4 -- [description](#)
regpatch: Samba-4.20.4 -- [description](#)

regshell: Samba-4.20.4 -- [description](#)
regtree: Samba-4.20.4 -- [description](#)
reindexdb: PostgreSQL-16.4 -- [description](#)
reordercap: Wireshark-4.2.6 -- [description](#)
replace: MariaDB-10.11.8 -- [description](#)
request-key: keyutils-1.6.3 -- [description](#)
rescan-scsi-bus.sh: sg3_utils-1.48 -- [description](#)
resize: xterm-393 -- [description](#)
resolveip: MariaDB-10.11.8 -- [description](#)
resolve_stack_dump: MariaDB-10.11.8 -- [description](#)
rgview: Vim-9.1.0660 -- [description](#)
ri: Ruby-3.3.4 -- [description](#)
ristretto: Ristretto-0.13.2 -- [description](#)
rjoe: JOE-4.6 -- [description](#)
rmiregistry: OpenJDK-22.0.2 -- [description](#)
rnano: Nano-8.1 -- [description](#)
rndc: BIND-9.20.0 -- [description](#)
rndc-confgen: BIND-9.20.0 -- [description](#)
roqet: Rasqlal-0.9.33 -- [description](#)
rotatelogs: Apache-2.4.62 -- [description](#)
route: Net-tools-2.10 -- [description](#)
rpc.mountd: NFS-Utils-2.6.4 -- [description](#)
rpc.nfsd: NFS-Utils-2.6.4 -- [description](#)
rpc.statd: NFS-Utils-2.6.4 -- [description](#)
rpcbind: rpcbind-1.2.7 -- [description](#)
rpcclient: Samba-4.20.4 -- [description](#)
rpcdebug: NFS-Utils-2.6.4 -- [description](#)
rpcgen: rpcsvc-proto-1.4.4 -- [description](#)
rpcinfo: rpcbind-1.2.7 -- [description](#)
rscsi: Cdrtools-3.02a09 -- [description](#)
rst2html: docutils-0.21.2 -- [description](#)
rst2html4: docutils-0.21.2 -- [description](#)
rst2html5: docutils-0.21.2 -- [description](#)
rst2latex: docutils-0.21.2 -- [description](#)
rst2man: docutils-0.21.2 -- [description](#)
rst2odt: docutils-0.21.2 -- [description](#)
rst2odt_prepstyles: docutils-0.21.2 -- [description](#)
rst2pseudoxml: docutils-0.21.2 -- [description](#)
rst2s5: docutils-0.21.2 -- [description](#)
rst2xetex: docutils-0.21.2 -- [description](#)
rst2xml: docutils-0.21.2 -- [description](#)
rstpep2html: docutils-0.21.2 -- [description](#)
rsvg-convert: librsvg-2.58.3 -- [description](#)
rsync: rsync-3.3.0 -- [description](#)
rsync-ssl: rsync-3.3.0 -- [description](#)
ruby: Ruby-3.3.4 -- [description](#)
run-with-aspell: Aspell-0.60.8.1 -- [description](#)
runant.pl: apache-ant-1.10.14 -- [description](#)
runant.py: apache-ant-1.10.14 -- [description](#)
rust-gdb: Rustc-1.80.1 -- [description](#)
rust-gdbgui: Rustc-1.80.1 -- [description](#)
rust=lldb: Rustc-1.80.1 -- [description](#)
rustc: Rustc-1.80.1 -- [description](#)
rustdoc: Rustc-1.80.1 -- [description](#)
rustfmt: Rustc-1.80.1 -- [description](#)
rvlc: VLC-3.0.21 -- [description](#)
sadf: Sysstat-12.7.6 -- [description](#)
saidar: libstatgrab-0.92.1 -- [description](#)
samba-gpupdate: Samba-4.20.4 -- [description](#)
samba-log-parser: Samba-4.20.4 -- [description](#)
samba-regedit: Samba-4.20.4 -- [description](#)
samba-tool: Samba-4.20.4 -- [description](#)
sancov: LLVM-18.1.7 -- [description](#)
sane-config: SANE-1.2.1 -- [description](#)
sane-find-scanner: SANE-1.2.1 -- [description](#)
saned: SANE-1.2.1 -- [description](#)
sanstats: LLVM-18.1.7 -- [description](#)
sar: Sysstat-12.7.6 -- [description](#)
saslauthd: Cyrus_SASL-2.1.28 -- [description](#)
sasldblistusers2: Cyrus_SASL-2.1.28 -- [description](#)
saslpasswd2: Cyrus_SASL-2.1.28 -- [description](#)
sassc: sassc-3.6.2 -- [description](#)
sbcdec: SBC-2.0 -- [description](#)
sbcenc: SBC-2.0 -- [description](#)
sbcinfo: SBC-2.0 -- [description](#)
sbiload: alsatools-1.2.11 -- [description](#)

scalar: Git-2.46.0 -- [description](#)
scan-build: LLVM-18.1.7 -- [description](#)
scan-build-py: LLVM-18.1.7 -- [description](#)
scan-view: LLVM-18.1.7 -- [description](#)
scanimate: SANE-1.2.1 -- [description](#)
sccmap: Graphviz-12.1.0 -- [description](#)
scgcheck: Cdrtools-3.02a09 -- [description](#)
sclient: MIT Kerberos V5-1.21.3 -- [description](#)
scmp_sys_resolver: libseccomp-2.5.5 -- [description](#)
scons: SCons-4.8.0 -- [description](#)
scons-configure-cache: SCons-4.8.0 -- [description](#)
sconsign: SCons-4.8.0 -- [description](#)
scour: Scour-0.38.2 -- [description](#)
scp: OpenSSH-9.8p1 -- [description](#)
screen: Screen-4.9.1 -- [description](#)
screengrab: screengrab-2.8.0 -- [description](#)
scsi_logging_level: sg3_utils-1.48 -- [description](#)
scsi_mandat: sg3_utils-1.48 -- [description](#)
scsi_readcap: sg3_utils-1.48 -- [description](#)
scsi_ready: sg3_utils-1.48 -- [description](#)
scsi_satl: sg3_utils-1.48 -- [description](#)
scsi_start: sg3_utils-1.48 -- [description](#)
scsi_stop: sg3_utils-1.48 -- [description](#)
scsi_temperature: sg3_utils-1.48 -- [description](#)
sddm: sddm-0.21.0 -- [description](#)
sddm-greeter: sddm-0.21.0 -- [description](#)
sdl-config: sdl12-compat-1.2.68 -- [description](#)
sdl2-config: SDL2-2.30.6 -- [description](#)
seahorse: Seahorse-43.0 -- [description](#)
seamonkey: Seamonkey-2.53.18.2 -- [description](#)
secret-tool: libsecret-0.21.4 -- [description](#)
sendmail: Postfix-3.9.0 -- [description](#)
sensord: sendmail-8.18.1 -- [description](#)
sensors: Lm-sensors-3-6-0 -- [description](#)
sensors-conf-convert: Lm-sensors-3-6-0 -- [description](#)
sensors-detect: Lm-sensors-3-6-0 -- [description](#)
serialver: Lm-sensors-3-6-0 -- [description](#)
sessreg: OpenJDK-22.0.2 -- [description](#)
setcifsacl: Xorg_Applications -- [description](#)
setpci: cifs-utils-7.0 -- [description](#)
setxkbmap: pciutils-3.13.0 -- [description](#)
sexp-conv: Xorg_Applications -- [description](#)
sfconvert: Nettle-3.10 -- [description](#)
sfddiff: AudioFile-0.3.6 -- [description](#)
sfdp: FontForge-20230101 -- [description](#)
sfinfo: Graphviz-12.1.0 -- [description](#)
sfreerdp: AudioFile-0.3.6 -- [description](#)
sfreerdp-server: FreeRDP-3.7.0 -- [description](#)
sftp: FreeRDP-3.7.0 -- [description](#)
sgdisk: OpenSSH-9.8p1 -- [description](#)
sginfo: gptfdisk-1.0.10 -- [description](#)
sgml2xml: sg3_utils-1.48 -- [description](#)
sgmldiff: OpenSP-1.5.2 -- [description](#)
sgmlnorm: DocBook-utils-0.6.14 -- [description](#)
sgmlspl: OpenSP-1.5.2 -- [description](#)
sgmlspl.pl: SGMLSpm-1.1 -- [description](#)
sgmlwhich: SGMLSpm-1.1 -- [description](#)
sgm_dd: sgml-common-0.6.3 -- [description](#)
sgp_dd: sg3_utils-1.48 -- [description](#)
sg_bg_ctl: sg3_utils-1.48 -- [description](#)
sg_compare_and_write: sg3_utils-1.48 -- [description](#)
sg_copy_results: sg3_utils-1.48 -- [description](#)
sg_dd: sg3_utils-1.48 -- [description](#)
sg_decode_sense: sg3_utils-1.48 -- [description](#)
sg_emc_trespass: sg3_utils-1.48 -- [description](#)
sg_format: sg3_utils-1.48 -- [description](#)
sg_get_config: sg3_utils-1.48 -- [description](#)
sg_get_elem_status: sg3_utils-1.48 -- [description](#)
sg_get_lba_status: sg3_utils-1.48 -- [description](#)
sg_ident: sg3_utils-1.48 -- [description](#)
sg_inq: sg3_utils-1.48 -- [description](#)
sg_logs: sg3_utils-1.48 -- [description](#)
sg_luns: sg3_utils-1.48 -- [description](#)
sg_map: sg3_utils-1.48 -- [description](#)
sg_map26: sg3_utils-1.48 -- [description](#)

sg_modes: sg3_utils-1.48 -- [description](#)
sg_opcodes: sg3_utils-1.48 -- [description](#)
sg_persist: sg3_utils-1.48 -- [description](#)
sg_prevent: sg3_utils-1.48 -- [description](#)
sg_raw: sg3_utils-1.48 -- [description](#)
sg_rbuf: sg3_utils-1.48 -- [description](#)
sg_rdac: sg3_utils-1.48 -- [description](#)
sg_read: sg3_utils-1.48 -- [description](#)
sg_readcap: sg3_utils-1.48 -- [description](#)
sg_read_attr: sg3_utils-1.48 -- [description](#)
sg_read_block_limits: sg3_utils-1.48 -- [description](#)
sg_read_buffer: sg3_utils-1.48 -- [description](#)
sg_read_long: sg3_utils-1.48 -- [description](#)
sg_reassign: sg3_utils-1.48 -- [description](#)
sg_referrals: sg3_utils-1.48 -- [description](#)
sg_rep_pip: sg3_utils-1.48 -- [description](#)
sg_rep_zones: sg3_utils-1.48 -- [description](#)
sg_requests: sg3_utils-1.48 -- [description](#)
sg_reset: sg3_utils-1.48 -- [description](#)
sg_reset_wp: sg3_utils-1.48 -- [description](#)
sg_rmsn: sg3_utils-1.48 -- [description](#)
sg_rtppg: sg3_utils-1.48 -- [description](#)
sg_safte: sg3_utils-1.48 -- [description](#)
sg_sanitize: sg3_utils-1.48 -- [description](#)
sg_sat_identify: sg3_utils-1.48 -- [description](#)
sg_sat_phy_event: sg3_utils-1.48 -- [description](#)
sg_sat_read_gplog: sg3_utils-1.48 -- [description](#)
sg_sat_set_features: sg3_utils-1.48 -- [description](#)
sg_scan: sg3_utils-1.48 -- [description](#)
sg_seek: sg3_utils-1.48 -- [description](#)
sg_senddiag: sg3_utils-1.48 -- [description](#)
sg_ses: sg3_utils-1.48 -- [description](#)
sg_ses_microcode: sg3_utils-1.48 -- [description](#)
sg_start: sg3_utils-1.48 -- [description](#)
sg_stpg: sg3_utils-1.48 -- [description](#)
sg_stream_ctl: sg3_utils-1.48 -- [description](#)
sg_sync: sg3_utils-1.48 -- [description](#)
sg_test_rwbuf: sg3_utils-1.48 -- [description](#)
sg_timestamp: sg3_utils-1.48 -- [description](#)
sg_turs: sg3_utils-1.48 -- [description](#)
sg_unmap: sg3_utils-1.48 -- [description](#)
sg_verify: sg3_utils-1.48 -- [description](#)
sg_vpd: sg3_utils-1.48 -- [description](#)
sg_write_buffer: sg3_utils-1.48 -- [description](#)
sg_write_long: sg3_utils-1.48 -- [description](#)
sg_write_same: sg3_utils-1.48 -- [description](#)
sg_write_verify: sg3_utils-1.48 -- [description](#)
sg_write_x: sg3_utils-1.48 -- [description](#)
sg_wr_mode: sg3_utils-1.48 -- [description](#)
sg_xcopy: sg3_utils-1.48 -- [description](#)
sg_zone: sg3_utils-1.48 -- [description](#)
shar: Sharutils-4.15.2 -- [description](#)
sharesec: Samba-4.20.4 -- [description](#)
sharkd: Wireshark-4.2.6 -- [description](#)
showmount: NFS-Utils-2.6.4 -- [description](#)
siggen: Tripwire-2.4.3.7 -- [description](#)
simple-scan: Simple-scan-46.0 -- [description](#)
simpleexpand: Xapian-1.4.26 -- [description](#)
simpleindex: Xapian-1.4.26 -- [description](#)
simplesearch: Xapian-1.4.26 -- [description](#)
simple_draw: libraw-0.21.2 -- [description](#)
sim_client: MIT Kerberos V5-1.21.3 -- [description](#)
sim_server: MIT Kerberos V5-1.21.3 -- [description](#)
skdump: libatasmart-0.19 -- [description](#)
sktest: libatasmart-0.19 -- [description](#)
slapacl: OpenLDAP-2.6.8 -- [description](#)
slapadd: OpenLDAP-2.6.8 -- [description](#)
slapauth: OpenLDAP-2.6.8 -- [description](#)
slapcat: OpenLDAP-2.6.8 -- [description](#)
slapd: OpenLDAP-2.6.8 -- [description](#)
slapdn: OpenLDAP-2.6.8 -- [description](#)
slapindex: OpenLDAP-2.6.8 -- [description](#)
slapmodify: OpenLDAP-2.6.8 -- [description](#)
slappasswd: OpenLDAP-2.6.8 -- [description](#)
slapschema: OpenLDAP-2.6.8 -- [description](#)
slaptest: OpenLDAP-2.6.8 -- [description](#)

slattach: [Net-tools-2.10](#) -- [description](#)
slencheck: [unixODBC-2.3.12](#) -- [description](#)
sliceprint: [Enscript-1.6.6](#) -- [description](#)
slsh: [slang-2.3.3](#) -- [description](#)
sm-notify: [NFS-Utils-2.6.4](#) -- [description](#)
smartctl: [smartmontools-7.4](#) -- [description](#)
smartd: [smartmontools-7.4](#) -- [description](#)
smartypants: [Smartypants-2.0.1](#) -- [description](#)
smb2-quota: [cifs-utils-7.0](#) -- [description](#)
smbcacls: [Samba-4.20.4](#) -- [description](#)
smbclient: [Samba-4.20.4](#) -- [description](#)
smbcontrol: [Samba-4.20.4](#) -- [description](#)
smbcquotas: [Samba-4.20.4](#) -- [description](#)
smbd: [Samba-4.20.4](#) -- [description](#)
smbget: [Samba-4.20.4](#) -- [description](#)
smbinfo: [cifs-utils-7.0](#) -- [description](#)
smbpasswd: [Samba-4.20.4](#) -- [description](#)
smbspool: [Samba-4.20.4](#) -- [description](#)
smbstatus: [Samba-4.20.4](#) -- [description](#)
smbtar: [Samba-4.20.4](#) -- [description](#)
smbtorture: [Samba-4.20.4](#) -- [description](#)
smbtree: [Samba-4.20.4](#) -- [description](#)
smime_keys: [Mutt-2.2.13](#) -- [description](#)
smproxy: [Xorg_Applications](#) -- [description](#)
smrsh: [sendmail-8.18.1](#) -- [description](#)
snapshot: [Snapshot-46.3](#) -- [description](#)
sndfile-cmp: [libsndfile-1.2.2](#) -- [description](#)
sndfile-concat: [libsndfile-1.2.2](#) -- [description](#)
sndfile-convert: [libsndfile-1.2.2](#) -- [description](#)
sndfile-deinterleave: [libsndfile-1.2.2](#) -- [description](#)
sndfile-info: [libsndfile-1.2.2](#) -- [description](#)
sndfile-interleave: [libsndfile-1.2.2](#) -- [description](#)
sndfile-metadata-get: [libsndfile-1.2.2](#) -- [description](#)
sndfile-metadata-set: [libsndfile-1.2.2](#) -- [description](#)
sndfile-play: [libsndfile-1.2.2](#) -- [description](#)
sndfile-salvage: [libsndfile-1.2.2](#) -- [description](#)
sntp: [ntp-4.2.8p18](#) -- [description](#)
solid-hardware6: [Building_KDE_Frameworks_6.5.0_\(KF6\)](#). -- [description](#)
soundstretch: [solid-6.5.0_for_lxqt](#) -- [description](#)
spa-acp-tool: [SoundTouch-2.3.3](#) -- [description](#)
spa-inspect: [Pipewire-1.2.3](#) -- [description](#)
spa-json-dump: [Pipewire-1.2.3](#) -- [description](#)
spa-monitor: [Pipewire-1.2.3](#) -- [description](#)
spa-resample: [Pipewire-1.2.3](#) -- [description](#)
spam: [OpenSP-1.5.2](#) -- [description](#)
spcat: [OpenSP-1.5.2](#) -- [description](#)
speaker-test: [alsa-utils-1.2.12](#) -- [description](#)
speexdec: [Speex-1.2.1](#) -- [description](#)
speexenc: [Speex-1.2.1](#) -- [description](#)
spell: [Aspell-0.60.8.1](#) -- [description](#)
spent: [OpenSP-1.5.2](#) -- [description](#)
sphinx-apidoc: [Sphinx-8.0.2](#) -- [description](#)
sphinx-autogen: [Sphinx-8.0.2](#) -- [description](#)
sphinx-build: [Sphinx-8.0.2](#) -- [description](#)
sphinx-quickstart: [Sphinx-8.0.2](#) -- [description](#)
spirv-as: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-cfg: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-dis: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-lesspipe.sh: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-link: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-lint: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-objdump: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-opt: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-reduce: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
spirv-val: [SPIRV-Tools-1.3.290.0](#) -- [description](#)
sqlite3: [SQLite-3.46.1](#) -- [description](#)
srptool: [GnuTLS-3.8.7.1](#) -- [description](#)
sscape_ctl: [alsa-tools-1.2.11](#) -- [description](#)
ssconvert: [Gnumeric-1.12.57](#) -- [description](#)
ssdiff: [Gnumeric-1.12.57](#) -- [description](#)
sserver: [MIT_Kerberos_V5-1.21.3](#) -- [description](#)
ssindex: [Gnumeric-1.12.57](#) -- [description](#)
ssh: [OpenSSH-9.8p1](#) -- [description](#)
ssh-add: [OpenSSH-9.8p1](#) -- [description](#)
ssh-agent: [OpenSSH-9.8p1](#) -- [description](#)

ssh-copy-id: OpenSSH-9.8p1 -- description
ssh-keygen: OpenSSH-9.8p1 -- description
ssh-keyscan: OpenSSH-9.8p1 -- description
sshd: OpenSSH-9.8p1 -- description
sshfs: sshfs-3.7.3 -- description
ssindex: Gnumeric-1.12.57 -- description
sst_sump: MariaDB-10.11.8 -- description
start-pulseaudio-x11: PulseAudio-17.0 -- description
start-statd: NFS-Utils-2.6.4 -- description
startfluxbox: Fluxbox-1.3.7 -- description
startlxqt: Ixqt-session-2.0.0 -- description
startx: xinit-1.4.2 -- description
startxfc4: xfce4-session-4.18.4 -- description
states: Enscript-1.6.6 -- description
statgrab: libstatgrab-0.92.1 -- description
statgrab-make-mrtg-config: libstatgrab-0.92.1 -- description
statgrab-make-mrtg-index: libstatgrab-0.92.1 -- description
stream: ImageMagick-7.1.1-36 -- description
stringify: JOE-4.6 -- description
stunnel: stunnel-5.72 -- description
stunnel3: stunnel-5.72 -- description
sudo: Sudo-1.9.15p5 -- description
sudoedit: Sudo-1.9.15p5 -- description
sudoku: FLTK-1.3.9 -- description
sudoreplay: Sudo-1.9.15p5 -- description
sudo_logsvrd: Sudo-1.9.15p5 -- description
sudo_sendlog: Sudo-1.9.15p5 -- description
suexec: Apache-2.4.62 -- description
svlc: VLC-3.0.21 -- description
svn: Subversion-1.14.3 -- description
svnadmin: Subversion-1.14.3 -- description
svnbench: Subversion-1.14.3 -- description
svndumpfilter: Subversion-1.14.3 -- description
svnfsfs: Subversion-1.14.3 -- description
svnlook: Subversion-1.14.3 -- description
svnmucc: Subversion-1.14.3 -- description
svnrdump: Subversion-1.14.3 -- description
svnserve: Subversion-1.14.3 -- description
svnsync: Subversion-1.14.3 -- description
svnversion: Subversion-1.14.3 -- description
swig: SWIG-4.2.1 -- description
sx: OpenSP-1.5.2 -- description
sxpm: Xorg_Libraries -- description
synclient: Xorg_Synaptics_Driver-1.9.2 -- description
syncqt.pl: qt5-components-5.15.14 -- description
syndaemon: Xorg_Synaptics_Driver-1.9.2 -- description
systemd-cryptenroll: Systemd-256.4 -- description
tab2space: tidy-html5-5.8.0 -- description
taglib-config: Taglib-2.0.1 -- description
tapestat: Sysstat-12.7.6 -- description
tcsh: Tcsh-6.24.13 -- description
tdbbackup: Samba-4.20.4 -- description
tdbdump: Samba-4.20.4 -- description
tdbrestore: Samba-4.20.4 -- description
tdbtool: Samba-4.20.4 -- description
tecla: Tecla-46.0 -- description
termidx: JOE-4.6 -- description
testparm: Samba-4.20.4 -- description
testpattern: Gutenprint-5.3.4 -- description
testsaslauthd: Cyrus_SASL-2.1.28 -- description
tex2xindy: xindy-2.5.1 -- description
texindy: xindy-2.5.1 -- description
TeX Live programs: texlive-20240312-source -- description
text2pcap: Wireshark-4.2.6 -- description
thunar: thunar-4.18.11 -- description
thunar-settings: thunar-4.18.11 -- description
thunar-volman: thunar-volman-4.18.0 -- description
thunar-volman-settings: thunar-volman-4.18.0 -- description
thunderbird: Thunderbird-128.1.0esr -- description
tickadj: ntp-4.2.8p18 -- description
tidy: tidy-html5-5.8.0 -- description
tiffcp: libtiff-4.6.0 -- description
tiffdump: libtiff-4.6.0 -- description
tiffinfo: libtiff-4.6.0 -- description
tiffset: libtiff-4.6.0 -- description
tiffsplit: libtiff-4.6.0 -- description

tifcc: Little CMS-2.16 -- [description](#)
time: Time-1.9 -- [description](#)
tjbench: libjpeg-turbo-3.0.1 -- [description](#)
TeX Live programs: install-tl- unix -- [description](#)
toc2cddb: Cdrdao-1.2.4 -- [description](#)
toc2cue: Cdrdao-1.2.4 -- [description](#)
toc2mp3: Cdrdao-1.2.4 -- [description](#)
touchpad-edge-detector: libevdev 1.13.2 -- [description](#)
tracegen: qt5-components-5.15.14 -- [description](#)
traceroute: Traceroute-2.1.5 -- [description](#)
traceroute6: Traceroute-2.1.5 -- [description](#)
tracker3: Tracker-3.7.3 -- [description](#)
tracker3-daemon: Tracker-miners-3.7.3 -- [description](#)
tracker3-endpoint: Tracker-3.7.3 -- [description](#)
tracker3-export: Tracker-3.7.3 -- [description](#)
tracker3-extract: Tracker-3.7.3 -- [description](#)
tracker3-help: Tracker-3.7.3 -- [description](#)
tracker3-import: Tracker-3.7.3 -- [description](#)
tracker3-index: Tracker-3.7.3 -- [description](#)
tracker3-info: Tracker-3.7.3 -- [description](#)
tracker3-reset: Tracker-3.7.3 -- [description](#)
tracker3-search: Tracker-3.7.3 -- [description](#)
tracker3-sparql: Tracker-3.7.3 -- [description](#)
tracker3-sql: Tracker-3.7.3 -- [description](#)
tracker3-status: Tracker-miners-3.7.3 -- [description](#)
tracker3-tag: Tracker-miners-3.7.3 -- [description](#)
transicc: Little CMS-2.16 -- [description](#)
translit: Lingua::Translit-0.29 -- [description](#)
transmission-create: Transmission-4.0.6 -- [description](#)
transmission-daemon: Transmission-4.0.6 -- [description](#)
transmission-edit: Transmission-4.0.6 -- [description](#)
transmission-gtk: Transmission-4.0.6 -- [description](#)
transmission-qt: Transmission-4.0.6 -- [description](#)
transmission-remote: Transmission-4.0.6 -- [description](#)
transmission-show: Transmission-4.0.6 -- [description](#)
tred: Graphviz-12.1.0 -- [description](#)
tree: tree-2.1.3 -- [description](#)
tripwire: Tripwire-2.4.3.7 -- [description](#)
trust: p11-kit-0.25.5 -- [description](#)
tshark: Wireshark-4.2.6 -- [description](#)
tsig-keygen: BIND-9.20.0 -- [description](#)
tumblerd: tumbler-4.18.2 -- [description](#)
twadmin: Tripwire-2.4.3.7 -- [description](#)
twm: twm-1.0.12 -- [description](#)
twopi: Graphviz-12.1.0 -- [description](#)
twprint: Tripwire-2.4.3.7 -- [description](#)
uchardet: Uchardet-0.0.8 -- [description](#)
uconv: icu-75.1 -- [description](#)
ucs2any: Xorg_Fonts -- [description](#)
udisksctl: UDisks-2.10.1 -- [description](#)
uic: qt5-components-5.15.14 -- [description](#)
umax_pp: SANE-1.2.1 -- [description](#)
umockdev-record: Umockdev-0.18.3 -- [description](#)
umockdev-run: Umockdev-0.18.3 -- [description](#)
umockdev-wrapper: Umockdev-0.18.3 -- [description](#)
umount.nfs: NFS-Utils-2.6.4 -- [description](#)
umount.nfs4: NFS-Utils-2.6.4 -- [description](#)
umount.udisks2: UDIsks-2.10.1 -- [description](#)
unbound: Unbound-1.21.0 -- [description](#)
unbound-anchor: Unbound-1.21.0 -- [description](#)
unbound-checkconf: Unbound-1.21.0 -- [description](#)
unbound-control: Unbound-1.21.0 -- [description](#)
unbound-control-setup: Unbound-1.21.0 -- [description](#)
unbound-host: Unbound-1.21.0 -- [description](#)
uncompface: Compface-1.5.2 -- [description](#)
unflatten: Graphviz-12.1.0 -- [description](#)
unifdef: unifdef-2.12 -- [description](#)
unifdefall: unifdef-2.12 -- [description](#)
uninstall_ndiff: Nmap-7.95 -- [description](#)
uninstall_zenmap: Nmap-7.95 -- [description](#)
uniproc: JOE-4.6 -- [description](#)
unix2dos: dos2unix-7.5.2 -- [description](#)
unix2mac: dos2unix-7.5.2 -- [description](#)
unix_chkpwd: Linux-PAM-1.6.1 -- [description](#)
unix_update: Linux-PAM-1.6.1 -- [description](#)
unopkg: LibreOffice-24.8.0 -- [description](#)

unprocessed_raw: libraw-0.21.2 -- [description](#)
unrar: UnRar-7.0.9 -- [description](#)
unshar: Sharutils-4.15.2 -- [description](#)
unzip: UnZip-6.0 -- [description](#)
unzipfsx: UnZip-6.0 -- [description](#)
update-ca-certificates: p11-kit-0.25.5 -- [description](#)
update-desktop-database: desktop-file-utils-0.27 -- [description](#)
update-leap: ntp-4.2.8p18 -- [description](#)
update-mime-database: shared-mime-info-2.4 -- [description](#)
update-smart-drive-db: smartmontools-7.4 -- [description](#)
upower: UPower-1.90.4 -- [description](#)
urxvt: rxvt-unicode-9.31 -- [description](#)
urxvtc: rxvt-unicode-9.31 -- [description](#)
urxvtd: rxvt-unicode-9.31 -- [description](#)
us428control: alsamixer-1.2.11 -- [description](#)
usb-devices: usbutils-017 -- [description](#)
usbhid-dump: usbutils-017 -- [description](#)
userdbctl: Systemd-256.4 -- [description](#)
usx2yloader: alsamixer-1.2.11 -- [description](#)
uuclient: MIT Kerberos V5-1.21.3 -- [description](#)
uudecode: Sharutils-4.15.2 -- [description](#)
uuencode: Sharutils-4.15.2 -- [description](#)
uuserver: MIT Kerberos V5-1.21.3 -- [description](#)
uxterm: xterm-393 -- [description](#)
v4l2-compliance: v4l-utils-1.28.1 -- [description](#)
v4l2-ctl: v4l-utils-1.28.1 -- [description](#)
v4l2-dbg: v4l-utils-1.28.1 -- [description](#)
v4l2-sysfs-path: v4l-utils-1.28.1 -- [description](#)
v4l2gl: v4l-utils-1.28.1 -- [description](#)
v4l2grab: v4l-utils-1.28.1 -- [description](#)
vacation: sendmail-8.18.1 -- [description](#)
vacuumdb: PostgreSQL-16.4 -- [description](#)
vacuumlo: PostgreSQL-16.4 -- [description](#)
vala-gen-introspect: Vala-0.56.17 -- [description](#)
valac: Vala-0.56.17 -- [description](#)
valadoc: Vala-0.56.17 -- [description](#)
valgrind: Valgrind-3.23.0 -- [description](#)
valgrind-di-server: Valgrind-3.23.0 -- [description](#)
valgrind-listener: Valgrind-3.23.0 -- [description](#)
vapigen: Vala-0.56.17 -- [description](#)
vcut: vorbis-tools-1.4.2 -- [description](#)
verify-uselistorder: LLVM-18.1.7 -- [description](#)
veritysetup: cryptsetup-2.7.4 -- [description](#)
vfat-resize: libblockdev-3.1.1 -- [description](#)
vgdb: Valgrind-3.23.0 -- [description](#)
vgimportclone: LVM2-2.03.26 -- [description](#)
vimdot: Graphviz-12.1.0 -- [description](#)
visualinfo: GLEW-2.2.0 -- [description](#)
visudo: Sudo-1.9.15p5 -- [description](#)
vlc: VLC-3.0.21 -- [description](#)
vlc-wrapper: VLC-3.0.21 -- [description](#)
vncconfig: Tigervnc-1.14.0 -- [description](#)
vncpasswd: Tigervnc-1.14.0 -- [description](#)
vncserver: Tigervnc-1.14.0 -- [description](#)
vncviewer: Tigervnc-1.14.0 -- [description](#)
vorbiscomment: vorbis-tools-1.4.2 -- [description](#)
vpxdec: libvpx-1.14.1 -- [description](#)
vpxenc: libvpx-1.14.1 -- [description](#)
vte-2.91: VTE-0.76.4 -- [description](#)
vte-2.91-gtk4: VTE-0.76.4 -- [description](#)
vwebp: libwebp-1.4.0 -- [description](#)
vxloader: alsamixer-1.2.11 -- [description](#)
watchgnupg: GnuPG-2.4.5 -- [description](#)
wayland-scanner: Wayland-1.23.0 -- [description](#)
wbinfo: Samba-4.20.4 -- [description](#)
WebKitWebDriver: WebKitGTK-2.44.3 -- [description](#)
webpinfo: libwebp-1.4.0 -- [description](#)
webpmux: libwebp-1.4.0 -- [description](#)
wget: Wget-1.24.5 -- [description](#)
which: Which-2.21 and Alternatives -- [description](#)
whiptail: newt-0.52.24 -- [description](#)
whois: Whois-5.4.3 -- [description](#)
winbindd: Samba-4.20.4 -- [description](#)
winpr-hash: FreeRDP-3.7.0 -- [description](#)
winpr-makecert: FreeRDP-3.7.0 -- [description](#)
wireplumber: Wireplumber-0.5.5 -- [description](#)

xfdesktop: Xfdesktop-4.18.1 -- [description](#)
xfdesktop-settings: Xfdesktop-4.18.1 -- [description](#)
xfpm-power-backlight-helper: xfce4-power-manager-4.18.4 -- [description](#)
xfreerdp: FreeRDP-3.7.0 -- [description](#)
xfsettingsd: xfce4-settings-4.18.6 -- [description](#)
xfs_admin: xfsprogs-6.9.0 -- [description](#)
xfs_bmap: xfsprogs-6.9.0 -- [description](#)
xfs_copy: xfsprogs-6.9.0 -- [description](#)
xfs_db: xfsprogs-6.9.0 -- [description](#)
xfs_estimate: xfsprogs-6.9.0 -- [description](#)
xfs_freeze: xfsprogs-6.9.0 -- [description](#)
xfs_fsr: xfsprogs-6.9.0 -- [description](#)
xfs_growfs: xfsprogs-6.9.0 -- [description](#)
xfs_info: xfsprogs-6.9.0 -- [description](#)
xfs_io: xfsprogs-6.9.0 -- [description](#)
xfs_logprint: xfsprogs-6.9.0 -- [description](#)
xfs_mdrestore: xfsprogs-6.9.0 -- [description](#)
xfs_metadump: xfsprogs-6.9.0 -- [description](#)
xfs_mkfile: xfsprogs-6.9.0 -- [description](#)
xfs_ncheck: xfsprogs-6.9.0 -- [description](#)
xfs_quota: xfsprogs-6.9.0 -- [description](#)
xfs_repair: xfsprogs-6.9.0 -- [description](#)
xfs_rtcp: xfsprogs-6.9.0 -- [description](#)
xfs_scrub: xfsprogs-6.9.0 -- [description](#)
xfs_scrub_all: xfsprogs-6.9.0 -- [description](#)
xfs_spaceman: xfsprogs-6.9.0 -- [description](#)
xfwm4: Xfwm4-4.18.0 -- [description](#)
xfwm4-settings: Xfwm4-4.18.0 -- [description](#)
xfwm4-tweaks-settings: Xfwm4-4.18.0 -- [description](#)
xfwm4-workspace-settings: Xfwm4-4.18.0 -- [description](#)
xgamma: Xorg_Applications -- [description](#)
xhost: Xorg_Applications -- [description](#)
xindy: xindy-2.5.1 -- [description](#)
xine: xine-ui-0.99.14 -- [description](#)
xine-bugreport: xine-ui-0.99.14 -- [description](#)
xine-check: xine-ui-0.99.14 -- [description](#)
xine-config: xine-lib-1.2.13 -- [description](#)
xine-list-1.2: xine-lib-1.2.13 -- [description](#)
xine-remote: xine-ui-0.99.14 -- [description](#)
xinit: xinit-1.4.2 -- [description](#)
xinput: Xorg_Applications -- [description](#)
xkbbell: Xorg_Applications -- [description](#)
xkcli: libxkbcommon-1.7.0 -- [description](#)
xkcomp: Xorg_Applications -- [description](#)
xkbevd: Xorg_Applications -- [description](#)
xkbvleds: Xorg_Applications -- [description](#)
xkbwatch: Xorg_Applications -- [description](#)
xkill: Xorg_Applications -- [description](#)
xlsatoms: Xorg_Applications -- [description](#)
xlsclients: Xorg_Applications -- [description](#)
xmessage: Xorg_Applications -- [description](#)
xml2-config: libxml2-2.13.3 -- [description](#)
xmcatalog: libxml2-2.13.3 -- [description](#)
xmlif: xmlto-0.0.29 -- [description](#)
xmllint: libxml2-2.13.3 -- [description](#)
xmlto: xmlto-0.0.29 -- [description](#)
xmodmap: Xorg_Applications -- [description](#)
Xnest: Xorg-Server-21.1.13 -- [description](#)
Xorg: Xorg-Server-21.1.13 -- [description](#)
xorrecord: libisoburn-1.5.6 -- [description](#)
xorriso: libisoburn-1.5.6 -- [description](#)
xorriso-dd-target: libisoburn-1.5.6 -- [description](#)
xorriso-tcltk: libisoburn-1.5.6 -- [description](#)
xorrisofs: libisoburn-1.5.6 -- [description](#)
xpr: Xorg_Applications -- [description](#)
xprop: Xorg_Applications -- [description](#)
xpstojpeg: libgxp-0.3.2 -- [description](#)
xpstodf: libgxp-0.3.2 -- [description](#)
xpstopng: libgxp-0.3.2 -- [description](#)
xpstops: libgxp-0.3.2 -- [description](#)
xpstosvg: libgxp-0.3.2 -- [description](#)
xrandr: Xorg_Applications -- [description](#)
xrdb: Xorg_Applications -- [description](#)
xrefresh: Xorg_Applications -- [description](#)
xscreensaver: XScreenSaver-6.09 -- [description](#)
xscreensaver-command: XScreenSaver-6.09 -- [description](#)

xscreensaver-demo: XScreenSaver-6.09 -- [description](#)
xscreensaver-settings: XScreenSaver-6.09 -- [description](#)
xset: Xorg_Applications -- [description](#)
xsetroot: Xorg_Applications -- [description](#)
xsetwacom: Xorg_Wacom_Driver-1.2.2 -- [description](#)
xsilt-config: libxslt-1.1.42 -- [description](#)
xsiltproc: libxslt-1.1.42 -- [description](#)
xtables-multi: iptables-1.8.10 -- [description](#)
xterm: xterm-393 -- [description](#)
Xvfb: Xorg_Server-21.1.13 -- [description](#)
xvinfo: Xorg_Applications -- [description](#)
Xvnc: Tigervnc-1.14.0 -- [description](#)
Xwayland: Xwayland-24.1.2 -- [description](#)
xwd: Xorg_Applications -- [description](#)
xwininfo: Xorg_Applications -- [description](#)
xwud: Xorg_Applications -- [description](#)
yapp: Parse::Yapp-1.21 -- [description](#)
yasm: yasm-1.3.0 -- [description](#)
yat2m: libgpg-error-1.50 -- [description](#)
yelp: Yelp-42.2 -- [description](#)
zip: Zip-3.0 -- [description](#)
zipcloak: Zip-3.0 -- [description](#)
zipgrep: UnZip-6.0 -- [description](#)
zipinfo: UnZip-6.0 -- [description](#)
zipnote: Zip-3.0 -- [description](#)
zipsplit: Zip-3.0 -- [description](#)
zsh: zsh-5.9 -- [description](#)
4channels: libraw-0.21.2 -- [description](#)

Libraries

ant-*jar: apache-ant-1.10.14 -- [description](#)
c-ares.so: c-ares-1.33.0 -- [description](#)
udev.so: dhcpcd-10.0.8 -- [description](#)
libduktape.so: duktape-2.7.0
evdev_drv.so: Xorg_Evdev_Driver-2.10.6 -- [description](#)
fop.jar: fop-2.9 -- [description](#)
libkeybinder-3.0.so: keybinder-3.0-0.3.2 -- [description](#)
liba52.so: Liba52-0.8.0 -- [description](#)
libaa.so: AAlib-1.4rc5 -- [description](#)
libabiword-3.0.so: AbiWord-3.0.5 -- [description](#)
libaccountsservice.so: AccountsService-23.13.9 -- [description](#)
libadwaita-1.so: libadwaita-1.5.3 -- [description](#)
libaio.so: libaio-0.3.113 -- [description](#)
libao.so: Libao-1.2.0 -- [description](#)
libaom.so: libaom-3.9.1 -- [description](#)
libappstream.so: AppStream-1.0.3 -- [description](#)
libappstream-glib.so: appstream-glib-0.8.3 -- [description](#)
libapr-1.so: Apr-1.7.4 -- [description](#)
libaprutil-1.so: Apr-Util-1.6.3 -- [description](#)
libarchive.so: libarchive-3.7.4 -- [description](#)
libasound.so: alsa-lib-1.2.12 -- [description](#)
libasound_module_pcm_a52.so: alsa-plugins-1.2.12 -- [description](#)
libasound_module_pcm_jack.so: alsa-plugins-1.2.12 -- [description](#)
libasound_module_pcm_oss.so: alsa-plugins-1.2.12 -- [description](#)
libasound_module_pcm_pulse.so: alsa-plugins-1.2.12 -- [description](#)
libasound_module_pcm_upmix.so: alsa-plugins-1.2.12 -- [description](#)
libasound_module_pcm_vdownmix.so: alsa-plugins-1.2.12 -- [description](#)
libasound_module_rate_samplerate.so: alsa-plugins-1.2.12 -- [description](#)
libaspell.so: Aspell-0.60.8.1 -- [description](#)
libass.so: libass-0.17.3 -- [description](#)
libassuan.so: libassuan-3.0.1 -- [description](#)
libatasmart.so: libatasmart-0.19 -- [description](#)
libatk-1.0.so: at-spi2-core-2.52.0 -- [description](#)
libatk-bridge.so: at-spi2-core-2.52.0 -- [description](#)
libatk-bridge-2.0.so: at-spi2-core-2.52.0 -- [description](#)
libatkmm-1.6.so: Atkmm-2.28.4 -- [description](#)
libatkmm-2.36.so: Atkmm-2.36.3 -- [description](#)
libatomic_ops.so: libatomic_ops-7.8.2 -- [description](#)
libatopology.so: alsa-lib-1.2.12 -- [description](#)
libatspi.so: at-spi2-core-2.52.0 -- [description](#)
libaudiofile.so: AudioFile-0.3.6 -- [description](#)
libavcodec.so: FFmpeg-7.0.2 -- [description](#)
libavdevice.so: FFmpeg-7.0.2 -- [description](#)
libavfilter.so: FFmpeg-7.0.2 -- [description](#)
libavformat.so: FFmpeg-7.0.2 -- [description](#)

[libavif.so](#):
[libavresample](#):
[libavutil.so](#):
[libbabl-0.1.so](#):
[libbluetooth.so](#):
[libbrasero-burn3.so](#):
[libbrasero-media3.so](#):
[libbrasero-utils3.so](#):
[libbrotlicommon{-,static.a,.so}](#):
[libbrotlidec{-,static.a,.so}](#):
[libbrotlienc{-,static.a,.so}](#):
[libbtparse.so](#):
[libburn.so](#):
[libbytesize.so](#):
[libcairo.so](#):
[libcairo-gobject.so](#):
[libcairo-script-interpreter.so](#):
[libcaiomm-1.0.so](#):
[libcaiomm-1.16.so](#):
[libcamel-1.2.so](#):
[libcanberra.so](#):
[libcanberra-gtk3.so](#):
[libcdda_interface](#):
[libcdda_paranoia](#):
[libcdio.so](#):
[libcdt.so](#):
[libcgraph.so](#):
[libcloudproviders.so](#):
[libcolord-gtk.so](#):
[libcolord-gtk4.so](#):
[libcolord.so](#):
[libcolordcompat.so](#):
[libcolordpriate.so](#):
[libcolorhug.so](#):
[libcompface.a](#):
[libcord.so](#):
[libcrack.so](#):
[libcups.so](#):
[libcupsfilters.so](#):
[libcurl.so](#):
[libdaemon.so](#):
[libdbus-glib-1.so](#):
[libdconf.so](#):
[libdevmapper.so](#):
[libdouble-conversion.so](#):
[libdrm.so](#):
[libdrm_amdgpu.so](#):
[libdrm_intel.so](#):
[libdrm_nouveau.so](#):
[libdrm_radeon.so](#):
[libdv.{so,a}](#):
[libdvdcss.so](#):
[libdvdnav.so](#):
[libdvdread.so](#):
[libebbackend-1.2.so](#):
[libebook-1.2.so](#):
[libebook-contacts-1.2.so](#):
[libecal-1.2.so](#):
[libecpg.{so,a}](#):
[libecpg_compat.{so,a}](#):
[libedata-book-1.2.so](#):
[libedata-cal-1.2.so](#):
[libedataserver-1.2.so](#):
[libedataserverui-3.0.so](#):
[libedataserverui-4.1.0.so](#):
[libefiboot.so](#):
[libefisec.so](#):
[libefivar.so](#):
[libei.so](#):
[libeis.so](#):
[libenchant-2.so](#):
[libepoxy.so](#):
[libetestserverutils.so](#):
[ebvdev.so](#):
[libexempi.so](#):
[libexif.so](#):

[Contents](#) -- [description](#)
[FFmpeg-7.0.2](#) -- [description](#)
[FFmpeg-7.0.2](#) -- [description](#)
[babl-0.1.108](#) -- [description](#)
[BlueZ-5.77](#) -- [description](#)
[Brasero-3.12.3](#) -- [description](#)
[Brasero-3.12.3](#) -- [description](#)
[brotli-1.1.0](#) -- [description](#)
[brotli-1.1.0](#) -- [description](#)
[brotli-1.1.0](#) -- [description](#)
[Text::BibTeX-0.89](#) -- [description](#)
[libburn-1.5.6](#) -- [description](#)
[libbytesize-2.11](#) -- [description](#)
[Cairo-1.18.0](#) -- [description](#)
[Cairo-1.18.0](#) -- [description](#)
[Cairo-1.18.0](#) -- [description](#)
[libcaiomm-1.0 \(caiomm-1.14.5\)](#) -- [description](#)
[libcaiomm-1.16 \(caiomm-1.18.0\)](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[libcanberra-0.30](#) -- [description](#)
[libcanberra-0.30](#) -- [description](#)
[CDParanoia-III-10.2](#) -- [description](#)
[CDParanoia-III-10.2](#) -- [description](#)
[libcdio-2.1.0](#) -- [description](#)
[Graphviz-12.1.0](#) -- [description](#)
[Graphviz-12.1.0](#) -- [description](#)
[libcloudproviders-0.3.6](#) -- [description](#)
[colord-gtk-0.3.1](#) -- [description](#)
[colord-gtk-0.3.1](#) -- [description](#)
[Colord-1.4.7](#) -- [description](#)
[Colord-1.4.7](#) -- [description](#)
[Colord-1.4.7](#) -- [description](#)
[Compface-1.5.2](#) -- [description](#)
[GC-8.2.6](#) -- [description](#)
[CrackLib-2.10.2](#) -- [description](#)
[Cups-2.4.10](#) -- [description](#)
[libcupsfilters-2.0.0](#) -- [description](#)
[cURL-8.9.1](#) -- [description](#)
[libdaemon-0.14](#) -- [description](#)
[dbus-glib-0.112](#) -- [description](#)
[DConf-0.40.0 / DConf-Editor-45.0.1](#) -- [description](#)
[LVM2-2.03.26](#) -- [description](#)
[Double-conversion-3.3.0](#) -- [description](#)
[Libdrm-2.4.122](#) -- [description](#)
[Libdrm-2.4.122](#) -- [description](#)
[Libdrm-2.4.122](#) -- [description](#)
[Libdrm-2.4.122](#) -- [description](#)
[Libdv-1.0.0](#) -- [description](#)
[libdvdcss-1.4.3](#) -- [description](#)
[Libdvdnav-6.1.1](#) -- [description](#)
[Libdvdread-6.1.3](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[PostgreSQL-16.4](#) -- [description](#)
[PostgreSQL-16.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[efivar-39](#) -- [description](#)
[efivar-39](#) -- [description](#)
[efivar-39](#) -- [description](#)
[libei-1.3.0](#) -- [description](#)
[libei-1.3.0](#) -- [description](#)
[enchant-2.8.2](#) -- [description](#)
[libepoxy-1.5.10](#) -- [description](#)
[evolution-data-server-3.52.4](#) -- [description](#)
[libevdev-1.13.2](#) -- [description](#)
[Exempi-2.6.5](#) -- [description](#)
[libexif-0.6.24](#) -- [description](#)

libexo-2.so: Exo-4.18.0 -- [description](#)
libexslt.so: libxslt-1.1.42 -- [description](#)
libfaac.so: FAAC-1_30 -- [description](#)
libfaad.so: FAAD2-2.11.1 -- [description](#)
libFalkonPrivate.so.3: falkon-24.08.0 -- [description](#)
libfdk-aac.so: fdk-aac-2.0.3 -- [description](#)
libfdt.so: dtc-1.7.1 -- [description](#)
libfftw3.so: fftw-3.3.10 -- [description](#)
libfftw3f.so: fftw-3.3.10 -- [description](#)
libfftw3f_threads.so: fftw-3.3.10 -- [description](#)
libfftw3l.so: fftw-3.3.10 -- [description](#)
libfftw3l_threads.so: fftw-3.3.10 -- [description](#)
libfftw3_threads.so: fftw-3.3.10 -- [description](#)
libFLAC{,++}.so: FLAC-1.4.3 -- [description](#)
libfltk.so: FLTK-1.3.9 -- [description](#)
libfm-extra.so: libfm-extra-1.3.2 -- [description](#)
libfm-qt6.so: libfm-qt-2.0.2 -- [description](#)
libfontconfig.so: Fontconfig-2.15.0 -- [description](#)
libfontenc.so: Xorg Libraries -- [description](#)
libfreerdp3.so: FreeRDP-3.7.0 -- [description](#)
libfreetype.so: FreeType-2.13.3 -- [description](#)
libfribidi.so: FriBidi-1.0.15 -- [description](#)
libFS.so: Xorg Libraries -- [description](#)
libfuse3.so: Fuse-3.16.2 -- [description](#)
libgailutil-3.so: GTK+-3.24.43 -- [description](#)
libgarcon-1.so: Garcon-4.18.2 -- [description](#)
libgavl.so: gavl-1.4.0 -- [description](#)
libgbm.so: Mesa-24.1.5 -- [description](#)
libgc.so: GC-8.2.6 -- [description](#)
libgccpp.so: GC-8.2.6 -- [description](#)
libgck-1.so: Gcr-3.41.2 -- [description](#)
libgck-2.so: Gcr-4.3.0 -- [description](#)
libgcr-4.so: Gcr-4.3.0 -- [description](#)
libgcrypt.so: libgcrypt-1.11.0 -- [description](#)
libgctba.so: GC-8.2.6 -- [description](#)
libgdata.so: libgdata-0.18.1 -- [description](#)
libgdk-3.so: GTK+-3.24.43 -- [description](#)
libgdkmm-3.0.so: Gtkmm-3.24.9 -- [description](#)
libgdkmm-4.0.so: Gtkmm-4.14.0 -- [description](#)
libgdk_pixbuf-2.0.so: gdk-pixbuf-2.42.12 -- [description](#)
libgedit-amtk-5.so: libgedit-amtk-5.8.0 -- [description](#)
libgee-0.8.so: libgee-0.20.6 -- [description](#)
libegl-0.4.so: gegl-0.4.48 -- [description](#)
libegl-npd-0.4.so: gegl-0.4.48 -- [description](#)
libgeocode-glib-2.so: geocode-glib-3.26.4 -- [description](#)
libgexiv2.so: gexiv2-0.14.3 -- [description](#)
libgif.so: giflib-5.2.2 -- [description](#)
libgimp-3.0.so: Gimp-20240711 -- [description](#)
libgimpbase-3.0.so: Gimp-20240711 -- [description](#)
libgimpcolor-3.0.so: Gimp-20240711 -- [description](#)
libgimpconfig-3.0.so: Gimp-20240711 -- [description](#)
libgimpmath-3.0.so: Gimp-20240711 -- [description](#)
libgimpmodule-2.0.so: Gimp-20240711 -- [description](#)
libgimpthumb-3.0.so: Gimp-20240711 -- [description](#)
libgimpui-3.0.so: Gimp-20240711 -- [description](#)
libgimpwidgets-3.0.so: Gimp-20240711 -- [description](#)
libgio-2.0.so: GLib-2.80.4 -- [description](#)
libgiomm-2.4.so: GLibmm-2.66.7 -- [description](#)
libgiomm-2.68.so: GLibmm-2.80.0 -- [description](#)
libgirepository-1.0.so: GLib-2.80.4 -- [description](#)
libgirepository-2.0.so: GLib-2.80.4 -- [description](#)
libgjs.so: Gjs-1.80.2 -- [description](#)
libEGL.so: Mesa-24.1.5 -- [description](#)
libGL.so: Mesa-24.1.5 -- [description](#)
libGLES2.so: Mesa-24.1.5 -- [description](#)
libGLESv1_CM.so: Mesa-24.1.5 -- [description](#)
libGLEW.so: GLEW-2.2.0 -- [description](#)
libglib-2.0.so: GLib-2.80.4 -- [description](#)
libglibmm-2.4.so: GLibmm-2.66.7 -- [description](#)
libglibmm-2.68.so: GLibmm-2.80.0 -- [description](#)
libglslang.so: Glslang-14.3.0 -- [description](#)
libGLU.so: GLU-9.0.3 -- [description](#)
libglut.so: Freeglut-3.6.0 -- [description](#)
libgmime-3.0.so: GMime-3.2.7 -- [description](#)
libgmodule-2.0.so: GLib-2.80.4 -- [description](#)
libgnome-autoar-0.so: gnome-autoar-0.4.4 -- [description](#)

libgnome-autoar-gtk-0.so: [gnome-autoar-0.4.4](#) -- [description](#)
libgnome-bluetooth-3.0.so: [gnome-bluetooth-46.1](#) -- [description](#)
libgnome-desktop-3.so: [gnome-desktop-44.1](#) -- [description](#)
libgnome-menu-3.so: [gnome-menus-3.36.0](#) -- [description](#)
libgnutls.so: [GnuTLS-3.8.7.1](#) -- [description](#)
libgoa-1.0.so: [gnome-online-accounts-3.50.4](#) -- [description](#)
libgoa-backend-1.0.so: [GLib-2.80.4](#) -- [description](#)
libgobject-2.0.so: [GOffice-0.10.57](#) -- [description](#)
libgoffice-0.10.so: [libgpg-error-1.50](#) -- [description](#)
libgpg-error.so: [GPGME-1.23.2](#) -- [description](#)
libgpgme.so: [GPGME-1.23.2](#) -- [description](#)
libgpgmepp.so: [GPM-1.20.7](#) -- [description](#)
libgpmm.so: [PostgreSQL-16.4](#) -- [description](#)
libgport.a: [Grantlee-5.3.1](#) -- [description](#)
libGrantlee_Templates.so: [Grantlee-5.3.1](#) -- [description](#)
libGrantlee_TextDocument.so: [Graphene-1.10.8](#) -- [description](#)
libgraphene-1.0.so: [Graphite2-1.3.14](#) -- [description](#)
libgraphite2.so: [Grilo-0.3.16](#) -- [description](#)
libgrilo.so: [Grilo-0.3.16](#) -- [description](#)
libgrinnet.so: [Grilo-0.3.16](#) -- [description](#)
libgripls.so: [ghostscript-10.03.1](#) -- [description](#)
libgsf-1.so: [libgsf-1.14.52](#) -- [description](#)
libgsl.so: [gsl-2.8](#) -- [description](#)
libgslcblas.so: [gsl-2.8](#) -- [description](#)
libgsound.so: [GSound-1.0.3](#) -- [description](#)
libgspell-1.so: [gspell-1.12.2](#) -- [description](#)
libgssapi_krb5.so: [MIT Kerberos V5-1.21.3](#) -- [description](#)
libgstbase-1.0.so: [gstreamer-1.24.7](#) -- [description](#)
libgstcheck-1.0.so: [gstreamer-1.24.7](#) -- [description](#)
libgstcontroller-1.0.so: [gstreamer-1.24.7](#) -- [description](#)
libgstnet-1.0.so: [gstreamer-1.24.7](#) -- [description](#)
libgststreamer-1.0.so: [gstreamer-1.24.7](#) -- [description](#)
libgthread-2.0.so: [GLib-2.80.4](#) -- [description](#)
libgtk-3.so: [GTK+-3.24.43](#) -- [description](#)
libgtk-4.so: [GTK-4.14.5](#) -- [description](#)
libgtk-vnc-2.0.so: [gtk-vnc-1.3.1](#) -- [description](#)
libgtkmm-3.0.so: [Gtkmm-3.24.9](#) -- [description](#)
libgtkmm-4.0.so: [Gtkmm-4.14.0](#) -- [description](#)
libgtksourcerview-3.0.so: [gtksourcerview-3.24.11](#) -- [description](#)
libgtksourcerview-4.so: [gtksourcerview4-4.8.4](#) -- [description](#)
libgtksourcerview-5.so: [gtksourcerview5-5.12.1](#) -- [description](#)
libgtop-2.0.so: [libgtop-2.41.3](#) -- [description](#)
libgucharmap_2_90.so: [Gucharmap-15.1.5](#) -- [description](#)
libgudev-1.0.so: [libgudev-238](#) -- [description](#)
libgusb.so: [libgusb-0.4.9](#) -- [description](#)
libgvfs.so: [Graphviz-12.1.0](#) -- [description](#)
libgvfscommon.so: [Gvfs-1.54.2](#) -- [description](#)
libgvnc-1.0.so: [gtk-vnc-1.3.1](#) -- [description](#)
libgvncpulse-1.0.so: [gtk-vnc-1.3.1](#) -- [description](#)
libgweather-4.so: [libgweather-4.4.2](#) -- [description](#)
libgxps.so: [libgxps-0.3.2](#) -- [description](#)
libhandle.so: [xfsprogs-6.9.0](#) -- [description](#)
libhandy-1.so: [libhandy-1.8.3](#) -- [description](#)
libharfbuzz.so: [harfBuzz-9.0.0](#) -- [description](#)
libharfbuzz-cairo: [harfBuzz-9.0.0](#) -- [description](#)
libharfbuzz-gobject.so: [harfBuzz-9.0.0](#) -- [description](#)
libharfbuzz-icu.so: [harfBuzz-9.0.0](#) -- [description](#)
libharfbuzz-subset.so: [harfBuzz-9.0.0](#) -- [description](#)
libhwy.so: [highway-1.2.0](#) -- [description](#)
libhwy_contrib.so: [highway-1.2.0](#) -- [description](#)
libhwy_test.so: [highway-1.2.0](#) -- [description](#)
libibus-1.0.so: [ibus-1.5.30](#) -- [description](#)
libical.so: [libical-3.0.18](#) -- [description](#)
libical-glib.so: [libical-3.0.18](#) -- [description](#)
libicalss.so: [libical-3.0.18](#) -- [description](#)
libicalcxx.so: [libical-3.0.18](#) -- [description](#)
libicalvcal.so: [libical-3.0.18](#) -- [description](#)
libical_cxx.so: [libical-3.0.18](#) -- [description](#)
libICE.so: [Xorg_Libraries](#) -- [description](#)
libicudata.so: [icu-75.1](#) -- [description](#)
libicui18n.so: [icu-75.1](#) -- [description](#)
libicuio.so: [icu-75.1](#) -- [description](#)
libicutest.so: [icu-75.1](#) -- [description](#)
libicutu.so: [icu-75.1](#) -- [description](#)
libicuuc.so: [icu-75.1](#) -- [description](#)

libid3.so: [id3lib-3.8.3](#) -- [description](#)
libidn.so: [libidn-1.42](#) -- [description](#)
libidn2.so: [libidn2-2.3.7](#) -- [description](#)
libigdgmm.so: [Intel-gmmlib-22.4.1](#) -- [description](#)
libigfxcmrt.so: [intel-media-24.2.5](#) -- [description](#)
libImlib2.so: [imlib2-1.12.3](#) -- [description](#)
libinh.so: [inih-58](#) -- [description](#)
libINIReader.so: [inih-58](#) -- [description](#)
libinkscape_base.so: [Inkscape-1.3.2](#) -- [description](#)
libinproctrace.so: [GDB-15.1](#) -- [description](#)
libinput.so: [libinput-1.26.1](#) -- [description](#)
libinput_drv.so: [Xorg_Libinput_Driver-1.4.0](#) -- [description](#)
libisoburn.so: [libisoburn-1.5.6](#) -- [description](#)
libisofs.so: [libisofs-1.5.6](#) -- [description](#)
libiw.so: [Wireless_Tools-29](#) -- [description](#)
libjansson.so: [Jansson-2.14](#) -- [description](#)
libjasper.so: [JasPer-4.2.4](#) -- [description](#)
libjavascriptcoregtk-4.1.so: [WebKitGTK-2.44.3](#) -- [description](#)
libjavascriptcoregtk-6.0.so: [WebKitGTK-2.44.3](#) -- [description](#)
libjpeg.so: [libjpeg-turbo-3.0.1](#) -- [description](#)
libjson-c.so: [JSON-C-0.17](#) -- [description](#)
libjson-glib-1.0.so: [JSON-GLib-1.8.0](#) -- [description](#)
libjxl.so: [libjxl-0.10.3](#) -- [description](#)
libjxl_cms.so: [libjxl-0.10.3](#) -- [description](#)
libjxl_extras_codec.so: [libjxl-0.10.3](#) -- [description](#)
libjxl_jni.so: [libjxl-0.10.3](#) -- [description](#)
libjxl_threads.so: [libjxl-0.10.3](#) -- [description](#)
libkadm5clnt.so: [MIT_Kerberos_V5-1.21.3](#) -- [description](#)
libkadm5srv.so: [MIT_Kerberos_V5-1.21.3](#) -- [description](#)
libkcddb.so: [libkcddb-24.08.0](#) -- [description](#)
libkdb5.so: [MIT_Kerberos_V5-1.21.3](#) -- [description](#)
libkeyutils.so: [keyutils-1.6.3](#) -- [description](#)
libKF5WaylandClient.so: [Contents](#)
libKF5WindowSystem.so: [Contents](#)
libKF6ConfigCore.so: [Kconfig-6.5.0_for_lxqt](#) -- [description](#)
libKF6ConfigGui.so: [Kconfig-6.5.0_for_lxqt](#) -- [description](#)
libKF6ConfigQml.so: [Kconfig-6.5.0_for_lxqt](#) -- [description](#)
libKF6IdleTime.so: [Contents](#)
libKF6Screen.so: [Contents](#)
libKF6ScreenDpms.so: [Contents](#)
libKF6solid.so: [Contents](#)
libkpathsea.so: [texlive-20240312-source](#) -- [description](#)
libkrad.so: [MIT_Kerberos_V5-1.21.3](#) -- [description](#)
libkrb5.so: [MIT_Kerberos_V5-1.21.3](#) -- [description](#)
libksba.{so,a}: [libksba-1.6.7](#) -- [description](#)
liblber.so: [OpenLDAP-2.6.8](#) -- [description](#)
liblcms2.so: [Little_CMS-2.16](#) -- [description](#)
libldap.so: [OpenLDAP-2.6.8](#) -- [description](#)
libldns.so: [Idns-1.8.4](#) -- [description](#)
liblibgedit-gtksourceview-300.so: [libgedit-gtksourceview-299.2.1](#) -- [description](#)
liblinear.so: [liblinear-247](#) -- [description](#)
libLLVMSPiRVLib.so: [SPIRV-LLVM-Translator-18.1.3](#) -- [description](#)
liblsof.so: [lsof-4.99.0](#) -- [description](#)
liblua.so: [Lua-5.4.7](#) -- [description](#)
liblua5.2.so: [Lua-5.2.4](#) -- [description](#)
liblxqt.so: [liblxqt-2.0.0](#) -- [description](#)
liblxqt-config-cursor.so: [lxqt-config-2.0.0](#) -- [description](#)
liblxqt-globalkeys-ui.so: [lxqt-globalkeys-2.0.0](#) -- [description](#)
liblz2.so: [LZO-2.10](#) -- [description](#)
libmad.so: [libmad-0.15.1b](#) -- [description](#)
libMagick++-7.Q16HDRI.so: [ImageMagick-7.1.1-36](#) -- [description](#)
libMagickCore-7.Q16HDRI.so: [ImageMagick-7.1.1-36](#) -- [description](#)
libMagickWand-7.Q16HDRI.so: [ImageMagick-7.1.1-36](#) -- [description](#)
libmbim-glib.so: [libmbim-1.26.4](#) -- [description](#)
libmenu-cache.so: [menu-cache-1.1.0](#) -- [description](#)
libmm-glib.so: [ModemManager-1.18.12](#) -- [description](#)
libmng.so: [libmng-2.0.3](#) -- [description](#)
libmnl.so: [libmnl-1.0.5](#) -- [description](#)
libmozs-115.so : [SpiderMonkey_from_firefox-115.14.0](#) -- [description](#)
libmp3lame.so: [LAME-3.100](#) -- [description](#)
libmp4v2.so: [FAAC-1_30](#) -- [description](#)
libmpeg2.so: [libmpeg2-0.5.1](#) -- [description](#)
libmpeg2convert.so: [libmpeg2-0.5.1](#) -- [description](#)
libmpg123.so: [mpg123-1.32.7](#) -- [description](#)
libmtdev.so: [mtdev-1.1.7](#) -- [description](#)
libmupdf.so: [MuPDF-1.24.8](#) -- [description](#)

libmusicbrainz5.so: [libmusicbrainz-5.1.0](#) -- [description](#)
libmutter-14.so: Mutter-46.4 -- [description](#)
libmutter-test-14.so: Mutter-46.4 -- [description](#)
libmypaint.so: libmypaint-1.6.1 -- [description](#)
libnautilus-extension.so: Nautilus-46.2 -- [description](#)
libneon.so: neon-0.33.0 -- [description](#)
libnetapi.so: Samba-4.20.4 -- [description](#)
libnghttp2.so: nghttp2-1.62.1 -- [description](#)
libnl*-3.so: libnl-3.10.0 -- [description](#)
libnm.so: NetworkManager-1.48.8 -- [description](#)
libnma.so: libnma-1.10.6 -- [description](#)
libnma-gtk4.so: libnma-1.10.6 -- [description](#)
libnotify.so: libnotify-0.8.3 -- [description](#)
libnpth.so: npth-1.7 -- [description](#)
libnsl.so: libnsl-2.0.1 -- [description](#)
libnspr4.so: NSPR-4.35 -- [description](#)
libnss_winbind.so: Samba-4.20.4 -- [description](#)
libnss_wins.so: Samba-4.20.4 -- [description](#)
libntfs-3g.so: ntfs-3g-2022.10.3 -- [description](#)
libnvme.so: libnvme-1.10 -- [description](#)
liboauth.so: liboauth-1.0.3 -- [description](#)
libobrender.so: openbox-3.6.1 -- [description](#)
libobt.so: openbox-3.6.1 -- [description](#)
liboffis.so: libe1-1.3.0 -- [description](#)
libogg.so: libogg-1.3.5 -- [description](#)
libopus.so: Opus-1.5.2 -- [description](#)
libosp.so: OpenSP-1.5.2 -- [description](#)
libout123.so: mpg123-1.32.7 -- [description](#)
libp11-kit.so: p11-kit-0.25.5 -- [description](#)
libpam.so: Linux-PAM-1.6.1 -- [description](#)
libpango-1.0.so: Pango-1.54.0 -- [description](#)
libpangomm-1.4.so: Pangomm-2.46.4 -- [description](#)
libpangomm-2.48.so: Pangomm-2.54.0 -- [description](#)
libpaper.so: libpaper-2.2.5 -- [description](#)
libparted.so: parted-3.6 -- [description](#)
libpathplan.so: Graphviz-12.1.0 -- [description](#)
pcap-config: libpcap-1.10.4 -- [description](#)
libpci.{a,so}: libpcap-1.10.4 -- [description](#)
libpci.so: pciutils-3.13.0 -- [description](#)
libpciaccess.so: Xorg Libraries -- [description](#)
libpeas-1.0.so: libpeas-1.36.0 -- [description](#)
libpeas-gtk-1.0.so: libpeas-1.36.0 -- [description](#)
libpgtypes.{so,a}: PostgreSQL-16.4 -- [description](#)
libpipewire-0.3.so: Pipewire-1.2.3 -- [description](#)
libpixbufloader-avif.so: libavif-1.1.1 -- [description](#)
libpixbufloader-jxl.so: libjxl-0.10.3 -- [description](#)
libpixbufloader-svg.so: librsvg-2.58.3 -- [description](#)
libpixbufloader-webp.so: webp-pixbuf-loader-0.2.7 -- [description](#)
libpixman-1.so: Pixman-0.43.4 -- [description](#)
libplacebo.so: libplacebo-7.349.0 -- [description](#)
libplc4.so: NSPR-4.35 -- [description](#)
libplds4.so: NSPR-4.35 -- [description](#)
libpng.so: libpng-1.6.43 -- [description](#)
libpolkit-agent-1.so: Polkit-125 -- [description](#)
libpolkit-gobject-1.so: Polkit-125 -- [description](#)
libpoppler.so: Poppler-24.08.0 -- [description](#)
libpoppler-cpp.so: Poppler-24.08.0 -- [description](#)
libpoppler-glib.so: Poppler-24.08.0 -- [description](#)
libpoppler-qt5.so: Poppler-24.08.0 -- [description](#)
libpoppler-qt6.so: Poppler-24.08.0 -- [description](#)
libpopt.so: Popt-1.19 -- [description](#)
libportal.so: libportal-0.7.1 -- [description](#)
libportal-gtk3.so: libportal-0.7.1 -- [description](#)
libportal-gtk4.so: libportal-0.7.1 -- [description](#)
libportal-qt5.so: libportal-0.7.1 -- [description](#)
libpostproc.so: FFmpeg-7.0.2 -- [description](#)
libpotrace.so: Potrace-1.16 -- [description](#)
libppd.so: libppd-2.0.0 -- [description](#)
libpq.{so,a}: PostgreSQL-16.4 -- [description](#)
libprotobuf.so: Protobuf-27.3 -- [description](#)
libprotobuf-c.c.so: Protobuf-c-1.5.0 -- [description](#)
libprotobuf-lite.so: Protobuf-27.3 -- [description](#)
libprotoc.so: Protobuf-27.3 -- [description](#)
libpsl.so: libpsl-0.21.5 -- [description](#)
libpspell.so: Aspell-0.60.8.1 -- [description](#)
libptexenc.so: texlive-20240312-source -- [description](#)

[libptytty.so](#): libptytty-2.0 -- description
[libpwquality.so](#): libpwquality-1.4.5 -- description
[libqalculate.so](#): libqalculate-5.2.0 -- description
[libqca-qt6.so](#): Qca-2.3.9 -- description
[libqgpgme.so](#): GPGME-1.23.2 -- description
[libqmi-glib.so](#): libqmi-1.30.8 -- description
[libqpdf.so](#): Qpdf-11.9.1 -- description
[libqrencode.so](#): qrencode-4.1.1 -- description
[libQt6Xdg.so](#): libqtxdg-4.0.0 -- description
[libQt6XdgIconLoader.so](#): libqtxdg-4.0.0 -- description
[libqtermwidget5.so](#): qtermwidget-2.0.1 -- description
[libqtlxqt.so](#): lxqt-qtplugin-2.0.0 -- description
[libraptor2.so](#): Raptor-2.0.16 -- description
[libraw.so](#): libraw-0.21.2 -- description
[librest-1.0.so](#): rest-0.9.1 -- description
[librest-extras-1.0.so](#): rest-0.9.1 -- description
[librsvg-2.so](#): librsvg-2.58.3 -- description
[libruby.so](#): Ruby-3.3.4 -- description
[libsane.so](#): SANE-1.2.1 -- description
[libsane-* .so](#): SANE-1.2.1 -- description
[libsasl2.so](#): Cyrus_SASL-2.1.28 -- description
[libsbc.so](#): SBC-2.0 -- description
[libscg.a](#): Cdrtools-3.02a09 -- description
[libSDL.so](#): sdl12-compat-1.2.68 -- description
[libSDL2.so](#): SDL2-2.30.6 -- description
[libseccomp.so](#): libseccomp-2.5.5 -- description
[libsecret-1.so](#): libsecret-0.21.4 -- description
[libsensors.so](#): Lm-sensors-3-6-0 -- description
[libserf-1.so](#): Serf-1.3.10 -- description
[libsgutils2.so](#): sg3_utils-1.48 -- description
[libshumate-1.0.so](#): libshumate-1.2.3 -- description
[libsigc-2.0.so](#): libsigc++-2.12.1 -- description
[libsigc-3.0.so](#): libsigc++-3.6.0 -- description
[libsigsegv.so](#): libsigsegv-2.14 -- description
[libslirp.so](#): libslirp-4.8.0 -- description
[libSM.so](#): Xorg_Libraries -- description
[libsmbclient.so](#): Samba-4.20.4 -- description
[libsndfile.so](#): libsndfile-1.2.2 -- description
[libSoundTouch.so](#): SoundTouch-2.3.3 -- description
[libsoup-2.4.so](#): libsoup-2.74.3 -- description
[libsoup-3.0.so](#): libsoup-2.74.3 -- description
[libsoup-gnome-2.4.so](#): libsoup-2.74.3 -- description
[libsp.so](#): OpenSP-1.5.2 -- description
[libspeex.so](#): Speex-1.2.1 -- description
[libspeexdsp.so](#): Speex-1.2.1 -- description
[libspiro.so](#): Libspiro-20220722 -- description
[libSPIRV.so](#): Glslang-14.3.0 -- description
[libSPIRV-Tools.so](#): SPIRV-Tools-1.3.290.0 -- description
[libspreadsheet.so](#): Gnumeric-1.12.57 -- description
[libSPVRemapper.so](#): Glslang-14.3.0 -- description
[libsqLite3.so](#): SQLite-3.46.1 -- description
[libssh2.so](#): libssh2-1.11.0 -- description
[libstartup-notification-1.so](#): startup-notification-0.12 -- description
[libstatgrab.so](#): libstatgrab-0.92.1 -- description
[libstd-<16-byte-hash>.so](#): Rustc-1.80.1 -- description
[libstunnel.so](#): stunnel-5.72 -- description
[libsvn_*-1.so](#): Subversion-1.14.3 -- description
[libswresample.so](#): FFmpeg-7.0.2 -- description
[libswscale.so](#): FFmpeg-7.0.2 -- description
[libsyn123.so](#): mpg123-1.32.7 -- description
[libsyntax.so](#): texlive-20240312-source -- description
[libsysstat-qt6.so](#): libsysstat-1.0.0 -- description
[libtalloc.so](#): Talloc-2.4.2 -- description
[libtasn1.so](#): libtasn1-4.19.0 -- description
[libtexlua52.so](#): texlive-20240312-source -- description
[libtexluajit.so](#): texlive-20240312-source -- description
[libthunarx-3.so](#): thunar-4.18.11 -- description
[libtidy.so](#): tidy-html5-5.8.0 -- description
[libtiff.so](#): libtiff-4.6.0 -- description
[libtiffxx.so](#): libtiff-4.6.0 -- description
[libtirpc.so](#): libtirpc-1.3.5 -- description
[libtk8.6.so](#): Tk-8.6.14 -- description
[libtotem-plparser.so](#): totem-pl-parser-3.26.6 -- description
[libtotem-plparser-mini.so](#): totem-pl-parser-3.26.6 -- description
[libtracker-sparql-3.0.so](#): Tracker-3.7.3 -- description
[libtumbler-1.so](#): tumbler-4.18.2 -- description

libuchardet.so: [Uchardet-0.0.8](#) -- [description](#)
libudisks2.so: [UDisks-2.10.1](#) -- [description](#)
libuhttplibmock-1.0.so: [uhttplibmock-0.11.0](#) -- [description](#)
libumockdev.so: [Umockdev-0.18.3](#) -- [description](#)
libunbound.so: [Unbound-1.21.0](#) -- [description](#)
libunistring.so: [libunistring-1.2](#) -- [description](#)
libupower-glib.so: [UPower-1.90.4](#) -- [description](#)
libusb-1.0.so: [libusb-1.0.27](#) -- [description](#)
libuv.so: [libuv-1.48.0](#) -- [description](#)
libva.so: [libva-2.22.0](#) -- [description](#)
libvala-0.56.so: [Vala-0.56.17](#) -- [description](#)
libvdpau.so: [libvdpau-1.5](#) -- [description](#)
libvdpau.so: [libvdpau-va-gl-0.4.2](#) -- [description](#)
libvorbis.so: [libvorbis-1.3.7](#) -- [description](#)
libvpx.so: [libvpx-1.14.1](#) -- [description](#)
libvte-2.91.so: [VTE-0.76.4](#) -- [description](#)
libvte-2.91-gtk4.so: [VTE-0.76.4](#) -- [description](#)
libvulkan.so: [Vulkan-Loader-1.3.294](#) -- [description](#)
libwacom.so: [libwacom-2.12.2](#) -- [description](#)
libwayland-client.so: [Wayland-1.23.0](#) -- [description](#)
libwayland-cursor.so: [Wayland-1.23.0](#) -- [description](#)
libwayland-egl.so: [Wayland-1.23.0](#) -- [description](#)
libwayland-server.so: [Wayland-1.23.0](#) -- [description](#)
libwbclient.so: [Samba-4.20.4](#) -- [description](#)
libwebkit2gtk-4.1.so: [WebKitGTK-2.44.3](#) -- [description](#)
libwebkitgtk-6.0.so: [WebKitGTK-2.44.3](#) -- [description](#)
libwebrtc.so: [libwebrtc-1.4.0](#) -- [description](#)
libwireplumber-0.5.so: [Wireplumber-0.5.5](#) -- [description](#)
libwireshark.so: [Wireshark-4.2.6](#) -- [description](#)
libwiretap.so: [Wireshark-4.2.6](#) -- [description](#)
libwnck-3.so: [libwnck-43.0](#) -- [description](#)
libwoff2common.so: [woff2-1.0.2](#) -- [description](#)
libwoff2enc.so: [woff2-1.0.2](#) -- [description](#)
libwoff2dec.so: [woff2-1.0.2](#) -- [description](#)
libwv-1.2.so: [wv-1.2.9](#) -- [description](#)
libX11.so: [Xorg_Libraries](#) -- [description](#)
libx264.so: [x264-20240812](#) -- [description](#)
libx265.so: [x265-3.6](#) -- [description](#)
libXau.so: [libXau-1.0.11](#) -- [description](#)
libXaw.so: [Xorg_Libraries](#) -- [description](#)
libXaw6.so: [Xorg_Libraries](#) -- [description](#)
libXaw7.so: [Xorg_Libraries](#) -- [description](#)
libxcb.so: [libxcb-1.17.0](#) -- [description](#)
libxcb-cursor.so: [xcb-util-cursor-0.1.4](#) -- [description](#)
libxcb-ewmh.so: [xcb-util-wm-0.4.2](#) -- [description](#)
libxcb-icccm.so: [xcb-util-wm-0.4.2](#) -- [description](#)
libxcb-image.so: [xcb-util-image-0.4.1](#) -- [description](#)
libxcb-keysyms.so: [xcb-util-keysyms-0.4.1](#) -- [description](#)
libxcb-render-util.so: [xcb-util-renderutil-0.3.10](#) -- [description](#)
libxcb-util.so: [xcb-util-0.4.1](#) -- [description](#)
libXcomposite.so: [Xorg_Libraries](#) -- [description](#)
libXcursor.so: [Xorg_Libraries](#) -- [description](#)
libxcvt.so: [libxcvt-0.1.2](#) -- [description](#)
libXdamage.so: [Xorg_Libraries](#) -- [description](#)
libXdmcp.so: [libXdmcp-1.1.5](#) -- [description](#)
libxdot.so: [Graphviz-12.1.0](#) -- [description](#)
libXext.so: [Xorg_Libraries](#) -- [description](#)
libxfce4kbd-private-2.so: [libfce4ui-4.18.6](#) -- [description](#)
libxfce4panel-2.0.so: [xfc4-panel-4.18.6](#) -- [description](#)
libxfce4ui-1.so: [libfce4ui-4.18.6](#) -- [description](#)
libxfce4util.so: [libfce4util-4.18.2](#) -- [description](#)
libxfconf-0.so: [Xfconf-4.18.3](#) -- [description](#)
libXfixes.so: [Xorg_Libraries](#) -- [description](#)
libXfont2.so: [Xorg_Libraries](#) -- [description](#)
libXft.so: [Xorg_Libraries](#) -- [description](#)
libXi.so: [Xorg_Libraries](#) -- [description](#)
libXine.so: [Xorg_Libraries](#) -- [description](#)
libXinerama.so: [xine-lib-1.2.13](#) -- [description](#)
libXkbcommon.so: [Xorg_Libraries](#) -- [description](#)
libXkbcommon-x11.so: [libXkbcommon-1.7.0](#) -- [description](#)
libXkbfile.so: [libXkbcommon-1.7.0](#) -- [description](#)
libXkbregistry.so: [Xorg_Libraries](#) -- [description](#)
libXklavier.so: [libXkbcommon-1.7.0](#) -- [description](#)
libxml2.so: [libXklavier-5.4](#) -- [description](#)
libxmlb.so: [libxml2-2.13.3](#) -- [description](#)
libXmu.so: [libxmlb-0.3.19](#) -- [description](#)
libXmu.so: [Xorg_Libraries](#) -- [description](#)

```

libXmuu.so: Xorg_Libraries -- description
libXpm.so: Xorg_Libraries -- description
libXpresent.so: Xorg_Libraries -- description
libXrandr.so: Xorg_Libraries -- description
libXrender.so: Xorg_Libraries -- description
libXRes.so: Xorg_Libraries -- description
libxshmfence.so: Xorg_Libraries -- description
libxslt.so: libxslt-1.1.42 -- description
libXss.so: Xorg_Libraries -- description
libXt.so: Xorg_Libraries -- description
libXtst.so: Xorg_Libraries -- description
libXv.so: Xorg_Libraries -- description
libxvidcore.so: XviD-1.3.7 -- description
libXvMC.so: Xorg_Libraries -- description
libXvMCW.so: Xorg_Libraries -- description
libXxf86dga.so: Xorg_Libraries -- description
libXxf86vm.so: Xorg_Libraries -- description
libyaml.so: libyaml-0.2.5 -- description
libyasm.a: yasm-1.3.0 -- description
libyelp.so: Yelp-42.2 -- description
log4cplus.so: log4cplus-2.1.1 -- description
lxqt-globalkeys.so: lxqt-globalkeys-2.0.0 -- description
modesetting_drv.so: Xorg-Server-21.1.13 -- description
mod_authz_svn.so: Subversion-1.14.3 -- description
mod_dav_svn.so: Subversion-1.14.3 -- description
muparser.so: muparser-2.3.4 -- description
p11-kit-proxy.so: p11-kit-0.25.5 -- description
pam_pwquality.so: libpwquality-1.4.5 -- description
pam_systemd.so: Systemd-256.4 -- description
synaptics_drv.so: Xorg_Synaptics Driver-1.9.2 -- description
wacom_drv.so: Xorg_Wacom Driver-1.2.2 -- description

```

Kernel Configuration

Advanced Network Bridge:

```

ALSA:
Automounter:
bluez:
Bridge Utilities:
BTRFS Programs:
bubblewrap:
Capturing network packets :
cifs-utils:
cryptsetup:
cups:
escptutil for usb printers:
FAT Support in Kernel:
Fuse 3:
gpm:
intel-media:
intel-vaapi-driver:
iptables:
iw:
JFS Utilities:
Kea:
keyutils (testing):
libevdev:
libinput:
Linux-PAM:
lm_sensors:
lsof (testing):
lvm2:
mdadm:
mesa:
NetworkManager (test):
NFS Utilities:
ntfs-3g:
NTFS read-write support in kernel:
parted (testing):
pm-utils:
polkit (testing):
Power-profiles-daemon:
qemu:
qtwebengine:
Scanning devices:
tracker-miners:

```

Advanced Network Setup -- description

```

alsa-lib-1.2.12 -- description
autofs-5.1.9 -- description
BlueZ-5.77 -- description
bridge-utils-1.7.1 -- description
btrfs-progs-6.10.1 -- description
Bubblewrap-0.9.0 -- description
Wireshark-4.2.6 -- description
cifs-utils-7.0 -- description
cryptsetup-2.7.4 -- description
Cups-2.4.10 -- description
Gutenprint-5.3.4 -- description
dosfstools-4.2 -- description
Fuse-3.16.2 -- description
GPM-1.20.7 -- description
intel-media-24.2.5 -- description
intel-vaapi-driver-2.4.1 -- description
iptables-1.8.10 -- description
iw-6.9 -- description
jfsutils-1.1.15 -- description
Kea 2.6.1 DHCP Server -- description
keyutils-1.6.3 -- description
libevdev 1.13.2 -- description
libinput-1.26.1 -- description
Linux-PAM-1.6.1 -- description
Lm-sensors-3-6-0 -- description
lsof-4.99.0 -- description
LVM2-2.03.26 -- description
mdadm-4.3 -- description
Mesa-24.1.5 -- description
NetworkManager-1.48.8 -- description
NFS-Utils-2.6.4 -- description
ntfs-3g-2022.10.3 -- description
ntfs-3g-2022.10.3 -- description
parted-3.6 -- description
pm-utils-1.4.1 -- description
Polkit-125 -- description
Power-profiles-daemon-0.21 -- description
qemu-9.0.2 -- description
QtWebEngine-6.7.2 -- description
SANE-1.2.1 -- description
Tracker-miners-3.7.3 -- description

```

UEFI:
upower:
Support for Host-side USB:
Wireless Tools:
wpa_supplicant:
XFS Programs:
xorg-server:
xorg-wacom-driver:

[Using GRUB to Set Up the Boot Process with UEFI](#) -- [description](#)
[UPower-1.90.4](#) -- [description](#)
[ibus-1.0.27](#) -- [description](#)
[Wireless Tools-29](#) -- [description](#)
[wpa_supplicant-2.11](#) -- [description](#)
[xfsprogs-6.9.0](#) -- [description](#)
[Xorg-Server-21.1.13](#) -- [description](#)
[Xorg_Wacom Driver-1.2.2](#) -- [description](#)

Configuration Files

\$PGDATA/pg_hba.conf:
\$PGDATA/pg_indent.con:
\$PGDATA/postgresql.conf:
~/.AbiSuite/templates/normal.awt: [AbiWord-3.0.5](#) -- [description](#)
~/.ant/ant.conf: [apache-ant-1.10.14](#) -- [description](#)
~/.antrc: [apache-ant-1.10.14](#) -- [description](#)
~/.asoundrc: [alsa-lib-1.2.12](#) -- [description](#)
~/.config/openbox/autostart: [openbox-3.6.1](#) -- [description](#)
~/.bashrc: [The Bash Shell Startup Files](#) -- [description](#)
~/.bash_logout: [The Bash Shell Startup Files](#) -- [description](#)
~/.bash_profile: [The Bash Shell Startup Files](#) -- [description](#)
~/.config/pulse: [PulseAudio-17.0](#) -- [description](#)
~/.config/user-dirs.dirs: [Xdg-user-dirs-0.18](#) -- [description](#)
~/.config/xdg-desktop-portal/portals.conf: [xdg-desktop-portal-1.18.2](#) -- [description](#)
~/.config/*: [MC-4.8.32](#) -- [description](#)
~/.cshdirs: [Tcsh-6.24.13](#) -- [description](#)
~/.cshrc: [Tcsh-6.24.13](#) -- [description](#)
~/.dircolors: [The Bash Shell Startup Files](#) -- [description](#)
~/.profile: [The Bash Shell Startup Files](#) -- [description](#)
~/.enchant: [enchant-2.8.2](#) -- [description](#)
~/.fetchmailrc: [Fetchmail-6.4.39](#) -- [description](#)
~/.fluxbox/init: [Fluxbox-1.3.7](#) -- [description](#)
~/.fluxbox/keys: [Fluxbox-1.3.7](#) -- [description](#)
~/.fluxbox/menu: [Fluxbox-1.3.7](#) -- [description](#)
~/.fonts: [Fontconfig-2.15.0](#) -- [description](#)
~/.foprc: [fop-2.9](#) -- [description](#)
~/.gitconfig: [Git-2.46.0](#) -- [description](#)
~/.gpm-root: [GPM-1.20.7](#) -- [description](#)
~/.hgrc: [Mercurial-6.8.1](#) -- [description](#)
~/.history: [Tcsh-6.24.13](#) -- [description](#)
~/.config/gtk-4.0/settings.ini: [GTK-4.14.5](#) -- [description](#)
~/.config/gtk-3.0/settings.ini: [GTK+-3.24.43](#) -- [description](#)
~/.icewm/keys: [IceWM-3.6.0](#) -- [description](#)
~/.icewm/menu: [IceWM-3.6.0](#) -- [description](#)
~/.icewm/preferences: [IceWM-3.6.0](#) -- [description](#)
~/.icewm/toolbar: [IceWM-3.6.0](#) -- [description](#)
~/.icewm/winoptions: [IceWM-3.6.0](#) -- [description](#)
~/.joerc: [JOE-4.6](#) -- [description](#)
~/.ldaprc: [OpenLDAP-2.6.8](#) -- [description](#)
~/.libao: [Libao-1.2.0](#) -- [description](#)
~/.links/*: [vorbis-tools-1.4.2](#) -- [description](#)
~/.login: [Links-2.30](#) -- [description](#)
~/.logout: [Tcsh-6.24.13](#) -- [description](#)
~/.mailrc: [Tcsh-6.24.13](#) -- [description](#)
~/.config/openbox/menu.xml: [mailx-12.5](#) -- [description](#)
~/.mime.types: [openbox-3.6.1](#) -- [description](#)
~/.muttrc: [Mutt-2.2.13](#) -- [description](#)
~/.my.cnf: [Mutt-2.2.13](#) -- [description](#)
~/.nanorc: [MariaDB-10.11.8](#) -- [description](#)
~/.ncftp/*: [Nano-8.1](#) -- [description](#)
~/.ogg123rc: [NcFTP-3.2.7](#) -- [description](#)
~/.pangorc: [vorbis-tools-1.4.2](#) -- [description](#)
~/.procmailrc: [Pango-1.54.0](#) -- [description](#)
~/.profile: [Procmail-3.24](#) -- [description](#)
~/.config/openbox/rc.xml: [Dash-0.5.12](#) -- [description](#)
~/.screenrc: [openbox-3.6.1](#) -- [description](#)
~/.slshrc: [Screen-4.9.1](#) -- [description](#)
~/.ssh/*: [slang-2.3.3](#) -- [description](#)
~/.subversion/config: [OpenSSH-9.8p1](#) -- [description](#)
~/.tcshrc: [SubVersion-1.14.3](#) -- [description](#)
~/.vimrc: [Tcsh-6.24.13](#) -- [description](#)
~/.wgetrc: [The /etc/vimrc and ~/.vimrc Files](#) -- [description](#)
~/.wireshark/*: [Vim-9.1.0660](#) -- [description](#)

~/.wgetrc: [Wget-1.24.5](#) -- [description](#)
~/.wireshark/*: [Wireshark-4.2.6](#) -- [description](#)

~/.xine/config:
~/.xinitrc:

~/xscreensaver:
/boot/grub/grub.cfg:

/etc/aliases:

/etc/ant/ant.conf:
/etc/asound.conf:
/etc/at.allow:
/etc/at.deny:
/etc/autofs/auto.master:
/etc/autofs/auto.misc:
/etc/autofs/auto.net:
/etc/bashrc:
/etc/bluetooth/main.conf:
/etc/csh.cshrc:
/etc/csh.login:
/etc/csh.logout:
/etc/cups/*:
/etc/dbus-1/session.conf:
/etc/dbus-1/system.d/*:
/etc/dbus-1/system.conf:
/etc/default/nfs-utils:
/etc/default/useradd:
/etc/dhcpcd/dhcpcd.conf:
/etc/dircolors:
/etc/dovecot/conf.d/*:
/etc/dovecot/dovecot.conf:
/etc/dovecot/local.conf:
/etc/exim.conf:
/etc/exportfs:
/etc/fcron.allow:
/etc/fcron.conf:
/etc/fcron.deny:
/etc/fonts/*:
/etc/fonts/conf.d/*:
/etc/fstab:
/etc/gdm/custom.conf:
/etc/gimp/3.0/*:
/etc/gitconfig:
/etc/gpm-root.conf:
/etc/group:
/etc/httpd/extr/*:
/etc/httpd/httpd.conf:
/etc/issue:
/etc/joe/jmacsrc:
/etc/joe/joerc:
/etc/joe/jpicorc:
/etc/joe/jstarrc:
/etc/joe/rjoerc:
/etc/kea/kea-dhcp-ddns.conf:
/etc/kea/kea-dhcp4.conf:
/etc/kea/kea-dhcp6.conf:
/etc/kea/kea-ctrl-agent.conf:
/etc/krb5.conf:
/etc/ld.so.conf:

/etc/libao.conf:

/etc/lightdm/lightdm.conf:
/etc/login.defs:

/etc/lvm/lvm.conf:
/etc/lynx/lynx.cfg:
/etc/mail/*:
/etc/make-ca.conf:
/etc/mercurial/hgrc:
/etc/mime.types:
/etc/Muttrc:
/etc/mysql/my.cnf:
/etc/nail.rc:
/etc/named.conf:

xine-ui-0.99.14 -- [description](#)
Fluxbox-1.3.7 -- [description](#)
IceWM-3.6.0 -- [description](#)
XScreenSaver-6.09 -- [description](#)
About Console Fonts -- [description](#)
Using GRUB to Set Up the Boot Process with UEFI -- [description](#)
Exim-4.98 -- [description](#)
Postfix-3.9.0 -- [description](#)
apache-ant-1.10.14 -- [description](#)
alsa-lib-1.2.12 -- [description](#)
at-3.2.5 -- [description](#)
at-3.2.5 -- [description](#)
autofs-5.1.9 -- [description](#)
autofs-5.1.9 -- [description](#)
autofs-5.1.9 -- [description](#)
The Bash Shell Startup Files -- [description](#)
BlueZ-5.77 -- [description](#)
Tcsh-6.24.13 -- [description](#)
Tcsh-6.24.13 -- [description](#)
Tcsh-6.24.13 -- [description](#)
Cups-2.4.10 -- [description](#)
dbus-1.14.10 -- [description](#)
dbus-1.14.10 -- [description](#)
dbus-1.14.10 -- [description](#)
NFS-Utils-2.6.4 -- [description](#)
Configuring for Adding Users
dhpcd-10.0.8 -- [description](#)
The Bash Shell Startup Files -- [description](#)
Dovecot-2.3.21.1 -- [description](#)
Dovecot-2.3.21.1 -- [description](#)
Dovecot-2.3.21.1 -- [description](#)
Exim-4.98 -- [description](#)
NFS-Utils-2.6.4 -- [description](#)
Fcron-3.2.1 -- [description](#)
Fcron-3.2.1 -- [description](#)
Fcron-3.2.1 -- [description](#)
Fontconfig-2.15.0 -- [description](#)
Fontconfig-2.15.0 -- [description](#)
NFS-Utils-2.6.4 -- [description](#)
GDM-46.2 -- [description](#)
Gimp-20240711 -- [description](#)
Git-2.46.0 -- [description](#)
GPM-1.20.7 -- [description](#)
About System Users and Groups
Apache-2.4.62 -- [description](#)
Apache-2.4.62 -- [description](#)
Customizing your Logon with /etc/issue
JOE-4.6 -- [description](#)
Kea 2.6.1 DHCP Server -- [description](#)
MIT Kerberos V5-1.21.3 -- [description](#)
qt5-components-5.15.14 -- [description](#)
Qt-6.7.2 -- [description](#)
KDE Frameworks 6 Pre-installation Configuration
Libao-1.2.0 -- [description](#)
vorbis-tools-1.4.2 -- [description](#)
lightdm-1.32.0 -- [description](#)
About System Users and Groups
Shadow-4.16.0 -- [description](#)
Shadow-4.16.0 -- [description](#)
LVM2-2.03.26 -- [description](#)
Lynx-2.9.2 -- [description](#)
sendmail-8.18.1 -- [description](#)
make-ca-1.14 -- [description](#)
Mercurial-6.8.1 -- [description](#)
Mutt-2.2.13 -- [description](#)
Mutt-2.2.13 -- [description](#)
MariaDB-10.11.8 -- [description](#)
mailx-12.5 -- [description](#)
BIND-9.20.0 -- [description](#)

/etc/namedb/pz/127.0.0.0:
/etc/namedb/root.hints:
/etc/nanorc:
/etc/ncftp.*:
/etc/NetworkManager/NetworkManager.conf: NetworkManager-1.48.8 -- description
/etc/ntp.conf:
/etc/openldap/ldap.conf:
/etc/openldap/slapd.conf:
/etc/openldap/slapd.d/*:
/etc/pam.conf:

/etc/pam.d/*:

/etc/pango/pangorc:
/etc/passwd:
/etc/xdg/openbox/autostart:
/etc/sensors3.conf:
/etc/xdg/openbox/menu.xml:
/etc/xdg/openbox/rc.xml:
/etc/gtk-3.0/settings.ini:
/etc/pear.conf:
/etc/php-fpm.conf:
/etc/php-fpm.d/www.conf:
/etc/php.ini:
/etc/postfix/*:
/etc/procmailrc:
/etc/profile:

/etc/profile.d:
/etc/profile.d/bash_completion.sh: The Bash Shell Startup Files -- description
/etc/profile.d/dircolors.sh: The Bash Shell Startup Files -- description
/etc/profile.d/extrapaths.sh: The Bash Shell Startup Files -- description
/etc/profile.d/i18n.sh: The Bash Shell Startup Files -- description
/etc/profile.d/readline.sh: The Bash Shell Startup Files -- description
/etc/profile.d/umask.sh: The Bash Shell Startup Files -- description
/etc/proftpd.conf: ProFTPD-1.3.8b -- description
/etc/pulse/client.conf: PulseAudio-17.0 -- description
/etc/pulse/daemon.conf: PulseAudio-17.0 -- description
/etc/pulse/default.pa: PulseAudio-17.0 -- description
/etc/request-key.conf: keyutils-1.6.3 -- description
/etc/request-key.d/*:
/etc/resolv.conf:

/etc/rndc.conf:
/etc/rsyncd.conf:
/etc/samba/smb.conf:
/etc/sane.d/*.*conf:
/etc/saslauthd.conf:
/etc/screenrc:
/etc/security/*:

/etc/security/access.conf:
/etc/security/limits.conf:
/etc/sgml/catalog:

/etc/sgml/sgml.conf:
/etc/skel/*:
/etc/slsn.rc:
/etcSMARTD.conf:
/etc/ssh/sshd_config:
/etc/ssh/ssh_config:
/etc/stunnel/stunnel.conf:
/etc/subversion/config:
/etc/sudoers:
/etc/sysconfig/autofs.conf:
/etc/sysconfig/sysstat:
/etc/sysconfig/sysstat.ioconf:
/etc/tripwire/*:
/etc/udev/rules.d:
/etc/unbound/unbound.conf:
/etc/unixODBC/*:
/etc/vimrc:

/etc/wgetrc:
/etc/wireshark.conf:

```

/etc/wpa_supplicant/wpa_supplicant-* .conf: wpa_supplicant-2.11 -- description
/etc/X11/app-defaults/XScreenSaver: XScreenSaver-6.09 -- description
/etc/xdg-desktop-portal/portals.conf: xdg-desktop-portal-1.18.2 -- description
/etc/xdg/user-dirs.conf: Xdg-user-dirs-0.18 -- description
/etc/xdg/user-dirs.defaults: Xdg-user-dirs-0.18 -- description
/etc/xml/catalog: docbook-xml-4.5 -- description
docbook-xsl-nons-1.79.2 -- description
docbook-xml-5.0 -- description
docbook-xml-5.1 -- description
docbook-xml-4.5 -- description
docbook-xml-5.0 -- description
docbook-xml-5.1 -- description
/etc/xml/catalog: docbook-xml-5.0 -- description
docbook-xml-5.1 -- description
docbook-xml-4.5 -- description
docbook-xml-5.0 -- description
docbook-xml-5.1 -- description
/etc/zsh/zlogin: zsh-5.9 -- description
/etc/zsh/zlogout: zsh-5.9 -- description
/etc/zsh/zprofile: zsh-5.9 -- description
/etc/zsh/zshenv: zsh-5.9 -- description
/etc/zsh/zshrc: zsh-5.9 -- description
/etc/lightdm/keys.conf: lightdm-1.32.0 -- description
/lib/firmware: About_Firmware
/etc/lightdm/lightdm-gtk-greeter.conf: lightdm-1.32.0 -- description
/etc/sddm.conf: sddm-0.21.0 -- description
/etc/lightdm/users.conf: lightdm-1.32.0 -- description
/usr/share/enchant/enchant.ordering: enchant-2.8.2 -- description
/usr/share/fontconfig/conf.avail/*: Fontconfig-2.15.0 -- description
/usr/share/fonts: Fontconfig-2.15.0 -- description
/usr/share/graphviz/config: Graphviz-12.1.0 -- description
/usr/share/metainfo/org.linuxfromscratch.lfs.xml : AppStream-1.0.3 -- description
/usr/share/gtk-4.0/settings.ini: GTK-4.14.5 -- description
/usr/share/xdg-desktop-portal/portals.conf: xdg-desktop-portal-1.18.2 -- description
/var/lib/alsa/asound.state: alsa-utils-1.2.12 -- description
/var/lib/krb5kdc/kdc.conf: MIT Kerberos V5-1.21.3 -- description
/etc/geoclue/conf.d/90-lfs-google.conf: GeoClue-2.7.1 -- description

```

Bootscripts

General Information:

acpid:
 apache:
 at:
 autofs:
 avahi:
 bind:
 bluetooth:
 cups:
cups-browsed:
 dhcpcd:
 dovecot:
 exim:
 fcron:
 gdm:
gitserve:
 gpm:
 iptables:
 krb5:
 lightdm:
ModemManager:
 mysql:
NetworkManager:
 nfs-client:
ntp.service:
 php:
 postfix:
postgresql:
 proftpd:
 samba:
 saslauthd:
 sddm:
sendmail:
 slapd:
 smartd:
 sshd:
 stunnel:
svnserve:
 sysstat:
 unbound:
 winbindd:

BLFS Systemd Units

acpid-2.0.34 -- description
 Apache-2.4.62 -- description
 at-3.2.5 -- description
 autofs-5.1.9 -- description
 Avahi-0.8 -- description
 BIND-9.20.0 -- description
 BlueZ-5.77 -- description
 Cups-2.4.10 -- description
 cups-browsed-2.0.1 -- description
 dhcpcd-10.0.8 -- description
 Dovecot-2.3.21.1 -- description
 Exim-4.98 -- description
 Fcron-3.2.1 -- description
 GDM-46.2 -- description
 Running_a_Git_Server -- description
 GPM-1.20.7 -- description
 iptables-1.8.10 -- description
 MIT_Kerberos_V5-1.21.3 -- description
 lightdm-1.32.0 -- description
ModemManager-1.18.12 -- description
 MariaDB-10.11.8 -- description
NetworkManager-1.48.8 -- description
 NFS_Utils-2.6.4 -- description
 ntp-4.2.8p18 -- description
 PHP-8.3.10 -- description
 Postfix-3.9.0 -- description
PostgreSQL-16.4 -- description
 ProFTPD-1.3.8b -- description
 Samba-4.20.4 -- description
 Cyrus_SASL-2.1.28 -- description
 sddm-0.21.0 -- description
 sendmail-8.18.1 -- description
 OpenLDAP-2.6.8 -- description
 smartmontools-7.4 -- description
 OpenSSH-9.8p1 -- description
 stunnel-5.7.2 -- description
 Running_a_Subversion_Server -- description
 Sysstat-12.7.6 -- description
 Unbound-1.21.0 -- description
 Samba-4.20.4 -- description

zenmap:

[Nmap-7.95 -- description](#)

Others

ALSA Description:

[ALSA](#)

Configuring Xorg:

[Xorg-7 Testing and Configuration](#)

hwdata:

[hwdata-0.385](#)

Image::Magick:

[ImageMagick-7.1.1-36 -- description](#)

KDE Frameworks Introduction:

[KDE Frameworks 6 Pre-installation Configuration](#)

kirigami-addons:

[kirigami-addons-1.4.0](#)

Library archive (.la) files:

[About Libtool Archive \(.la\) files](#)

libraries: static or shared:

[Libraries: Static or shared?](#)

LXQt-post-install:

[LXQt Desktop Final Instructions](#)

LXQt Pre-Install:

[LXQt Desktop Pre-Install Instructions](#)

SGML DTD files:

[docbook-3.1-dtd -- description](#)

[docbook-4.5-dtd -- description](#)

[sgml-common-0.6.3 -- description](#)

[docbook-3.1-dtd -- description](#)

[docbook-4.5-dtd -- description](#)

[Tuning Fontconfig](#)

[About Console Fonts -- description](#)

[Vulnerabilities](#)

[docbook-xml-4.5 -- description](#)

[docbook-xml-5.0 -- description](#)

[sgml-common-0.6.3 -- description](#)

[docbook-xml-4.5 -- description](#)

[docbook-xml-4.5 -- description](#)

[docbook-xml-5.0 -- description](#)

[docbook-xml-5.1 -- description](#)

[docbook-xml-5.0 -- description](#)

[docbook-xml-5.1 -- description](#)

[docbook-xsl-nons-1.79.2 -- description](#)

XML entity files:

XML MOD files:

XML Schema files:

XML Schematron rules:

XSL Stylesheets: