

lfs:commands.org

idsyr

January 8, 2025

Contents

1	Preparing the Host System	5
1.1	Creating a New Partition	5
1.2	Creating a File System on the Partition	5
1.3	Setting The \$LFS Variable	6
1.4	Mounting the New Partition	6
2	Packages and Patches	6
3	Final Preparations	6
3.1	Creating a Limited Directory Layout in the LFS Filesystem	6
3.2	Adding the LFS User	6
3.3	Setting Up the Environment	7
4	Compiling a Cross-Toolchain	8
4.1	Binutils	8
4.2	GCC	9
4.3	Linux API Headers	11
4.4	Glibc	11
4.5	libstd++	12
5	Cross Compiling Temporary Tools	13
5.1	M4	13
5.2	Ncurses	13
5.3	Bash	14
5.4	Coreutils	14
5.5	Diffutils	15
5.6	file	15
5.7	findutils	16

5.8	gawk	16
5.9	grep	17
5.10	gzip	17
5.11	make	17
5.12	patch	18
5.13	sed	18
5.14	tar	19
5.15	xz	19
5.16	binutils	20
5.17	gcc	20
6	Entering Chroot and Building Additional Temporary Tools	22
6.1	Changing Ownership	22
6.2	Preparing Virtual Kernel File Systems	22
6.3	Mounting and Populating /dev	22
6.4	Mounting Virtual Kernel File Systems	22
6.5	Entering the Chroot Environment	23
6.6	Creating Directories	23
6.7	Creating Essential Files and Symlinks	24
6.8	gettext	25
6.9	bison	25
6.10	perl	26
6.11	python	26
6.12	Texinfo	27
6.13	util-linux	27
6.14	Cleaning up and Saving the Temporary System	28
6.15	Backup	28
6.16	Restore	28
7	Installing Basic System Software	29
7.1	man-pages	29
7.2	iana-etc	29
7.3	glibc	29
8	Configuring glibc	31
8.1	Adding nsswitch.conf	31
8.2	Adding Time Zone Data	31
8.3	Configuring the Dynamic Loader	32
8.4	zlib	32
8.5	bzip	32

8.6	xz	33
8.7	lz4	34
8.8	zstd	34
8.9	file	34
8.10	readline	35
8.11	m4	35
8.12	bc	36
8.13	flex	36
8.14	tcl	37
8.15	expect	38
8.16	dejaGNU	38
8.17	pkgconf	39
8.18	binutils	39
8.19	gmp	40
8.20	mpfr	41
8.21	mpc	42
8.22	attr	42
8.23	acl	43
8.24	libcap	43
8.25	libxcrypt	43
8.26	shadow	44
8.27	gcc	45
8.28	ncurses	47
8.29	sed	48
8.30	psmisc	48
8.31	gettext	49
8.32	bison	49
8.33	grep	50
8.34	bash	50
8.35	libtool	51
8.36	gdbm	51
8.37	gperf	52
8.38	expat	52
8.39	inetutils	52
8.40	less	53
8.41	perl	54
8.42	xml::parser	54
8.43	intltool	55
8.44	autoconf	55
8.45	automake	56

8.46	openssl	56
8.47	kmod	57
8.48	libelf	57
8.49	libffi	58
8.50	python	58
8.51	flit _{core}	59
8.52	setuptools	59
8.53	ninja	60
8.54	meson	60
8.55	coreutils	61
8.56	check	61
8.57	diffutils	62
8.58	gawk	62
8.59	findutils	63
8.60	groff	63
8.61	grub	64
8.62	gzip	64
8.63	iproute	64
8.64	kbd	65
8.65	libpipeline	65
8.66	make	66
8.67	patch	66
8.68	tar	67
8.69	texinfo	67
8.70	vim	68
8.71	MarkupSafe	69
8.72	jinja	69
8.73	udev	69
8.74	procps-ng	71
8.75	util-linux	71
8.76	e2fsprogs	72
8.77	sysklogd	73
8.78	sysvinit	74
8.79	stripping skipped FIXME	75

9 System Configuration 75

9.1	lfs-bootscripts	75
9.2	Creating Custom Udev Rules	75
9.3	CD _{ROM} Symlinks	75
9.4	Duplicate devices	75

10 Network Configuration	75
10.1 Creating Network Interface configuration files	75
10.2 Creating the /etc/resolv.conf	76
10.3 configuring the system hostname	76
10.4 Customizing the /etc/hosts	76
11 System V Bootscript Usage and Configuration	76
11.1 Configuring SysVinit	76
11.2 Configuring the System Clock	77
11.3 Linux Console	77
11.4 rc.site	78
11.5 System locale	80
11.6 /etc/inputrc	80
11.7 /etc/shells	81
12 Making the LFS System Bootable	81
12.1 Creating the /etc/fstab file	81
12.2 linux	82
12.3 Configuring Linux Module load order	82
12.4 GRUB	83
13 USEFUL	83
13.1	83

1 Preparing the Host System

1.1 Creating a New Partition

```
fdisk /dev/sdb
    g w
    n t w
# EFI system partition 512M
# Linux swap          4G
# Linux x86-64 root    32G
```

1.2 Creating a File System on the Partition

```
mkfs.fat -F32 /dev/sdb1
mkswap /dev/sdb2
mkfs -v -t ext4 /dev/sdb3
lsblk -f
```

1.3 Setting The \$LFS Variable

```
echo "LFS=/mnt/lfs" >> ~/.bashrc
bash
```

1.4 Mounting the New Partition

```
mkdir -pv $LFS
mount -v -t ext4 /dev/sdb3 $LFS
```

2 Packages and Patches

```
#!/bin/sh
mkdir -v $LFS/sources
chmod -v a+wt $LFS/sources      # only the owner of a file can delete the file within a
chown root:root $LFS/sources/* # otherwise files will be left owned by an unnamed UID
wget --input-file=wget-list-lfs-sysv --continue --directory-prefix=$LFS/sources
```

3 Final Preparations

3.1 Creating a Limited Directory Layout in the LFS Filesystem

```
#!/bin/sh

mkdir -pv $LFS/{etc,var} $LFS/usr/{bin,lib,sbin} # creating dirs
for i in bin lib sbin; do
    ln -sv usr/$i $LFS/$i # symbolic links in $LFS on dirs in usr/
done

case $(uname -m) in
    x86_64) mkdir -pv $LFS/lib64 ;; # if 64 -> +lib64
    esac

mkdir -pv $LFS/tools # for cross-comiler
```

3.2 Adding the LFS User

```
#!/bin/sh
```

```

groupadd lfs
useradd -s /bin/bash -g lfs -m -k /dev/null lfs

# -s /bin/bash - this makes bash the default for user lfs
# -g lfs - adds user lfs to group lfs
# -m - this creates a home directory for lfs
# -k /dev/null - this parametr prevents possible copting of files from skeleton direct
# lfs - this is the name of the new user

# for switch to lfs from non-root user (required pswd):
passwd lfs

# Grant lfs full access to all the directories under $LFS by making lfs the owner:
chown -v lfs $LFS/{usr{,/*},lib,var,etc,bin,sbin,tools}
case $(uname -m) in
    x86_64) chown -v lfs $LFS/lib64 ;;
esac

# Start the shell as a login shell with an environment similar to a real login:
su - lfs

```

3.3 Setting Up the Environment

- while logged as lfs

```

#!/bin/sh

# The exec env -i.../bin/bash command in the .bash_profile file replaces the running s
cat > ~/.bash_profile << "EOF"
exec env -i HOME=$HOME TERM=$TERM PS1='\u:\w\$ ' /bin/bash
EOF

cat > ~/.bashrc << "EOF"
set +h
umask 022
LFS=/mnt/lfs
LC_ALL=POSIX
LFS_TGT=$(uname -m)-lfs-linux-gnu
PATH=/usr/bin

```

```

if [ ! -L /bin ]; then PATH=/bin:$PATH; fi
PATH=$LFS/tools/bin:$PATH
CONFIG_SITE=$LFS/usr/share/config.site
export LFS LC_ALL LFS_TGT PATH CONFIG_SITE
EOF
# set +h - turns off bash's hash function
# umask 022 - new files with 644 mode
# LFS=/mnt/lfs - $LFS
# LC_ALL=POSIX - controls the localization of certain programs
# LFS_TGT=$(uname -m)-lfs-linux-gnu - for cross compiler
# PATH=/usr/bin - std PATH variable
# if [ ! -L /bin ]; then PATH=/bin:$PATH; fi - if /bin is not a symbolic link, it must
# PATH=$LFS/tools/bin:$PATH - by putting ahead of the std path, the cross-compiler is
# CONFIG_SITE=$LFS/usr/share/config.site - prevent potential contamination from the ho
# export - visible within any sub-shells

[ ! -e /etc/bash.bashrc ] || mv -v /etc/bash.bashrc /etc/bash.bashrc.NOUSE
# disable instantiation of /etc/bash.bashrc to the initialization of bash

cat >> ~/.bashrc << "EOF"
export MAKEFLAGS=-j$(nproc)
EOF

```

4 Compiling a Cross-Toolchain

4.1 Binutils

- linker, assembler, other tools for handling object files

```

cd $LFS/sources
LFS_TARGET=binutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

mkdir build
cd build

../configure \
# install in tools dir
--prefix=$LFS/tools \

```



```

# look the target system libs in $LFS
--with-sysroot=$LFS \
# adjust binutil's build system for building cross linker
--target=$LFS_TGT \
# disable internationalization as i18n is not needed for temp tools
--disable-nls \
# disable profiler
--enable-gprofng=no \
# prevent the build from stopping on warnings
--disable-werror \
# runpath instead of rpath
--enable-new-dtags \
# only one hash table for shared libs
--enable-default-hash-style=gnu

```

```

make
make install

```

4.2 GCC

```

#!/bin/sh

```

```

cd $LFS/sources

```

```

tar -xf mpfr*tar*
mv mpfr*/ mpfr/

```

```

tar -xf gmp*tar*
mv gmp*/ gmp/

```

```

tar -xf mpc*tar*
mv mpc*/ mpc/

```

```

LFS_TARGET=gcc
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

```

```

# set default directory name for 64-bit libs to "lib"

```

```

case $(uname -m) in
    x86_64)
        sed -e '/m64=/s/lib64/lib/' \
            -i.orig gcc/config/i386/t-linux64
        ;;
esac

mkdir build
cd build

../configure \
--target=$LFS_TGT \
--prefix=$LFS/tools \
--with-glibc-version=2.40 \
--with-sysroot=$LFS \
--with-newlib \
--without-headers \
--enable-default-pie \
--enable-default-ssp \
--disable-nls \
--disable-shared \
--disable-multilib \
--disable-threads \
--disable-libatomic \
--disable-libgomp \
--disable-libquadmath \
--disable-libssp \
--disable-libvtv \
--disable-libstdcxx \
--enable-languages=c,c++

make
make install

cd ..
cat gcc/limitx.h gcc/glimits.h gcc/limity.h > \
'dirname $(LFS_TGT-gcc -print-libgcc-file-name)'/include/limits.h

```

4.3 Linux API Headers

```
#!/bin/sh
cd $LFS/sources
LFS_TARGET=linux
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

make mrproper

make headers
find usr/include -type f ! -name '*.h' -delete
cp -rv usr/include $LFS/usr
```

4.4 Glibc

```
#!/bin/sh
cd $LFS/sources
LFS_TARGET=glibc
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

case $(uname -m) in
    i?86)
        ln -sfv ld-linux.so.2 $LFS/lib/ld-lsb.so.3
        ;;
    x86_64)
        ln -sfv ../lib/ld-linux-x86-64.so.2 $LFS/lib64
        ln -sfv ../lib/ld-linux-x86-64.so.2 $LFS/lib64/ld-lsb-x86-64.so.3
        ;;
esac

mkdir build
cd build

echo "rootsbindir=/usr/sbin" > configparms

../configure \
--prefix=/usr \
--host=$LFS_TGT \
```

```

--build=$(../scripts/config.guess) \
--enable-kernel=4.19 \
--with-headers=$LFS/usr/include \
--disable-nscd \
libc_cv_slibdir=/usr/lib

make
make DESTDIR=$LFS install

sed '/RTLDLIST=/s@/usr@@g' -i $LFS/usr/bin/ldd

```

4.5 libstd++

```

#!/bin/sh
cd $LFS/sources
LFS_TARGET=gcc
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

mkdir build
cd build

../configure \
--prefix=$LFS/tools \
--with-sysroot=$LFS \
--target=$LFS_TGT \
--disable-nls \
--enable-gprofng=no \
--disable-werror \
--enable-new-dtags \
--enable-default-hash-style=gnu

make
make DESTDIR=$LFS install
rm -v $LFS/usr/lib/lib{stdc++{,exp,fs},supc++}.la

```

5 Cross Compiling Temporary Tools

5.1 M4

5.2 Ncurses

```
#!/bin/sh
```

```
cd $LFS/sources
LFS_TARGET=ncurses
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/
```

```
mkdir build
pushd build
    ../configure
    make -C include
    make -C progs tic
popd
```

```
./configure \
--prefix=/usr \
--host=$LFS_TGT \
--build=$(./config.guess) \
--mandir=/usr/share/man \
--with-manpage-format=normal \
--with-shared \
--without-normal \
--with-cxx-shared \
--without-debug \
--without-ada \
--disable-stripping
```

```
make
make DESTDIR=$LFS TIC_PATH=$(pwd)/build/progs/tic install
ln -sv libncursesw.so $LFS/usr/lib/libncurses.so
sed -e 's/^#if.*XOPEN.*$/#if 1/' \
    -i $LFS/usr/include/curses.h
```

5.3 Bash

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=bash
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--build=$(sh support/config.guess) \
--host=$LFS_TGT \
--without-bash-malloc \
bash_cv_strtold_broken=no

make
make DESTDIR=$LFS install
ln -sv bash $LFS/bin/sh
```

5.4 Coreutils

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=coreutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--host=$LFS_TGT \
--build=$(build-aux/config.guess) \
--enable-install-program=hostname \
--enable-no-install-program=kill,uptime

make
make DESTDIR=$LFS install

mv -v $LFS/usr/bin/chroot $LFS/usr/sbin
```

```

mkdir -pv $LFS/usr/share/man/man8
mv -v $LFS/usr/share/man/man1/chroot.1 $LFS/usr/share/man/man8/chroot.8
sed -i 's/"1"/"8"/' $LFS/usr/share/man/man8/chroot.8

```

5.5 Diffutils

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=diffutils
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--host=$LFS_TGT \
--build=$(./build-aux/config.guess)

make
make DESTDIR=$LFS install

```

5.6 file

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=file
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

mkdir build
pushd build
    ../configure \
    --disable-bzlib \
    --disable-libseccomp \
    --disable-xzlib \
    --disable-zlib
    make
popd

```

```
./configure --prefix=/usr --host=$LFS_TGT --build=$(./config.guess)
```

```
make FILE_COMPILE=$(pwd)/build/src/file
make DESTDIR=$LFS install
rm -v $LFS/usr/lib/libmagic.la
```

5.7 findutils

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=findutils
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--localstatedir=/var/lib/locate \
--host=$LFS_TGT \
--build=$(build-aux/config.guess)

make
make DESTDIR=$LFS install
```

5.8 gawk

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=gawk
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

sed -i 's/extras//' Makefile.in

./configure \
--prefix=/usr \
--host=$LFS_TGT \
--build=$(build-aux/config.guess)
```



```
make
make DESTDIR=$LFS install
```

5.9 grep

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=grep
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--host=$LFS_TGT \
--build=$(./build-aux/config.guess)

make
make DESTDIR=$LFS install
```

5.10 gzip

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=gzip
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure --prefix=/usr --host=$LFS_TGT
make
make DESTDIR=$LFS install
```

5.11 make

```
#!/bin/sh
cd $LFS/sources
LFS_TARGET=make
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/
```

```
./configure \  
--prefix=/usr \  
--without-guile \  
--host=$LFS_TGT \  
--build=$(build-aux/config.guess)
```

```
make  
make DESTDIR=$LFS install
```

5.12 patch

```
#!/bin/sh  
  
cd $LFS/sources  
LFS_TARGET=patch  
tar -xf $LFS_TARGET.tar*  
cd $LFS_TARGET*/  
  
./configure \  
--prefix=/usr \  
--host=$LFS_TGT \  
--build=$(build-aux/config.guess)  
  
make  
make DESTDIR=$LFS install
```

5.13 sed

```
#!/bin/sh  
  
cd $LFS/sources  
LFS_TARGET=sed  
tar -xf $LFS_TARGET.tar*  
cd $LFS_TARGET*/  
  
./configure \  
--prefix=/usr \  
--host=$LFS_TGT \  
--build=$(./build-aux/config.guess)
```

```
make
make DESTDIR=$LFS install
```

5.14 tar

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=tar
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr \
--host=$LFS_TGT \
--build=$(build-aux/config.guess)

make
make DESTDIR=$LFS install
```

5.15 xz

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=xz
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--host=$LFS_TGT \
--build=$(build-aux/config.guess) \
--disable-static \
--docdir=/usr/share/doc/xz-5.6.2

make
make DESTDIR=$LFS install
rm -v $LFS/usr/lib/liblzma.la
```

5.16 binutils

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=binutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

sed '6009s/$add_dir//' -i ltmain.sh

mkdir build
cd build

../configure \
--prefix=/usr \
--build=$(../config.guess) \
--host=$LFS_TGT \
--disable-nls \
--enable-shared \
--enable-gprofng=no \
--disable-werror \
--enable-64-bit-bfd \
--enable-new-dtags \
--enable-default-hash-style=gnu

make
make DESTDIR=$LFS install
rm -v $LFS/usr/lib/lib{bfd,ctf,ctf-nobfd,opcodes,sframe}.{a,la}

cd ..
```

5.17 gcc

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=mpfr
tar -xf $LFS_TARGET*tar*
mv $LFS_TARGET*/ $LFS_TARGET/
```

```

LFS_TARGET=gmp
tar -xf $LFS_TARGET*tar*
mv $LFS_TARGET*/ $LFS_TARGET/

LFS_TARGET=mpc
tar -xf $LFS_TARGET*tar*
mv $LFS_TARGET*/ $LFS_TARGET/

LFS_TARGET=gcc
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

case $(uname -m) in
    x86_64)
        sed -e '/m64=/s/lib64/lib/' \
            -i.orig gcc/config/i386/t-linux64
        ;;
esac

sed '/thread_header =/s/@.*@/gthr-posix.h/' \
    -i libgcc/Makefile.in libstdc++-v3/include/Makefile.in

mkdir build
cd build

../configure \
--build=$(../config.guess) \
--host=$LFS_TGT \
--target=$LFS_TGT \
LDFLAGS_FOR_TARGET=-L$PWD/$LFS_TGT/libgcc \
--prefix=/usr \
--with-build-sysroot=$LFS \
--enable-default-pie \
--enable-default-ssp \
--disable-nls \
--disable-multilib \
--disable-libatomic \
--disable-libgomp \
--disable-libquadmath \

```

```
--disable-libsanitizer \
--disable-libssp \
--disable-libvtv \
--enable-languages=c,c++ \
```

```
make
make DESTDIR=$LFS install
ln -sv gcc $LFS/usr/bin/cc

cd ..
```

<=====

6 Entering Chroot and Building Additional Temporary Tools

6.1 Changing Ownership

- to root

```
chown --from lfs -R root:root $LFS/{usr,lib,var,etc,bin,sbin,tools}
case $(uname -m) in
    x86_64)
        chown --from lfs -R root:root $LFS/lib64
    ;;
esac
```

6.2 Preparing Virtual Kernel File Systems

```
mkdir -pv $LFS/{dev,proc,sys,run}
```

6.3 Mounting and Populating /dev

```
mount -v --bind /dev $LFS/dev
```

6.4 Mounting Virtual Kernel File Systems

```
mount -vt devpts devpts -o gid=5,mode=0620 $LFS/dev/pts
mount -vt proc proc $LFS/proc
mount -vt sysfs sysfs $LFS/sys
mount -vt tmpfs tmpfs $LFS/run
```

```

if [ -h $LFS/dev/shm ]; then
    install -v -d -m 1777 $LFS$(realpath /dev/shm)
else
    mount -vt tmpfs -o nosuid,nodev tmpfs $LFS/dev/shm
fi

```

6.5 Entering the Chroot Environment

```

chroot "$LFS" /usr/bin/env -i \
HOME=/root \
TERM="$TERM" \
PS1='(lfs chroot) \u:\w\$ ' \
PATH=/usr/bin:/usr/sbin \
MAKEFLAGS="-j$(nproc)" \
TESTSUITEFLAGS="-j$(nproc)" \
/bin/bash --login

```

6.6 Creating Directories

```

mkdir -pv /{boot,home,mnt,opt,srv}

mkdir -pv /etc/{opt,sysconfig}
mkdir -pv /lib/firmware
mkdir -pv /media/{floppy,cdrom}
mkdir -pv /usr/{,local/}{include,src}
mkdir -pv /usr/lib/locale
mkdir -pv /usr/local/{bin,lib,sbin}
mkdir -pv /usr/{,local/}share/{color,dict,doc,info,locale,man}
mkdir -pv /usr/{,local/}share/{misc,terminfo,zoneinfo}
mkdir -pv /usr/{,local/}share/man/man{1..8}
mkdir -pv /var/{cache,local,log,mail,opt,spool}
mkdir -pv /var/lib/{color,misc,locate}
ln -sfv /run /var/run
ln -sfv /run/lock /var/lock
install -dv -m 0750 /root
install -dv -m 1777 /tmp /var/tmp

```

6.7 Creating Essential Files and Symlinks

```
ln -sv /proc/self/mounts /etc/mtab
```

```
cat > /etc/hosts << EOF
127.0.0.1 localhost $(hostname)
::1
    localhost
EOF
```

```
cat > /etc/passwd << "EOF"
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/dev/null:/usr/bin/false
daemon:x:6:6:Daemon User:/dev/null:/usr/bin/false
messagebus:x:18:18:D-Bus Message Daemon User:/run/dbus:/usr/bin/false
uidd:x:80:80:UUID Generation Daemon User:/dev/null:/usr/bin/false
nobody:x:65534:65534:Unprivileged User:/dev/null:/usr/bin/false
EOF
```

```
cat > /etc/group << "EOF"
root:x:0:
bin:x:1:daemon
sys:x:2:
kmem:x:3:
tape:x:4:
tty:x:5:
daemon:x:6:
floppy:x:7:
disk:x:8:
lp:x:9:
dialout:x:10:
audio:x:11:
video:x:12:
utmp:x:13:
cdrom:x:15:
adm:x:16:
messagebus:x:18:
input:x:24:
mail:x:34:
kvm:x:61:
uidd:x:80:
```



```
wheel:x:97:
users:x:999:
nogroup:x:65534:
EOF
```

```
localedef -i C -f UTF-8 C.UTF-8
```

```
echo "tester:x:101:101::/home/tester:/bin/bash" >> /etc/passwd
echo "tester:x:101:" >> /etc/group
install -o tester -d /home/tester
```

```
exec /usr/bin/bash --login
```

```
touch /var/log/{btmp,lastlog,faillog,wtmp}
chgrp -v utmp /var/log/lastlog
chmod -v 664 /var/log/lastlog
chmod -v 600 /var/log/btmp
```

6.8 gettext

```
#!/bin/sh\n\n\n
```

```
cd $LFS/sources
LFS_TARGET=gettext
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/
```

```
./configure --disable-shared
make
cp -v gettext-tools/src/{msgfmt,msgmerge,xgettext} /usr/bin
```

6.9 bison

```
#!/bin/sh
```

```
cd $LFS/sources
LFS_TARGET=bison
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/
```

```
./configure \  
--prefix=/usr \  
--docdir=/usr/share/doc/bison-3.8.2
```

```
make  
make install
```

6.10 perl

```
#!/bin/sh  
  
#+INCLUDE: "install_addnltool_perl.sh" src shell  
cd $LFS/sources  
LFS_TARGET=perl  
tar -xf $LFS_TARGET*tar*  
cd $LFS_TARGET*/  
  
sh Configure -des \  
-D prefix=/usr \  
-D vendorprefix=/usr \  
-D useshrplib \  
-D privlib=/usr/lib/perl5/5.40/core_perl \  
-D archlib=/usr/lib/perl5/5.40/core_perl \  
-D sitelib=/usr/lib/perl5/5.40/site_perl \  
-D sitearch=/usr/lib/perl5/5.40/site_perl \  
-D vendorlib=/usr/lib/perl5/5.40/vendor_perl \  
-D vendorarch=/usr/lib/perl5/5.40/vendor_perl  
  
make  
make install
```

6.11 python

```
#!/bin/sh  
  
#+INCLUDE: "install_addnltool_python.sh" src shell  
cd $LFS/sources  
LFS_TARGET=python  
tar -xf $LFS_TARGET*tar*
```

```

cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--enable-shared \
--without-ensurepip

make
make install

```

6.12 Texinfo

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=texinfo
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
make install

```

6.13 util-linux

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=util-linux
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

mkdir -pv /var/lib/hwclock

./configure \
--libdir=/usr/lib \
--runstatedir=/run \
--disable-chfn-chsh \
--disable-login \

```

```

--disable-nologin \
--disable-su \
--disable-setpriv \
--disable-runuser \
--disable-pylibmount \
--disable-static \
--disable-liblastlog2 \
--without-python \
ADJTIME_PATH=/var/lib/hwclock/adjtime \
--docdir=/usr/share/doc/util-linux-2.40.2

make
make install

```

6.14 Cleaning up and Saving the Temporary System

```

rm -rf /usr/share/{info,man,doc}/*
find /usr/{lib,libexec} -name \*.la -delete
# rm -rf /tools

```

6.15 Backup

```

exit

mountpoint -q $LFS/dev/shm && umount $LFS/dev/shm
umount $LFS/dev/pts
umount $LFS/{sys,proc,run,dev}

cd $LFS
tar -cJpf $HOME/lfs-temp-tools-12.2.tar.xz .

```

6.16 Restore

```

cd $LFS
rm -rf ./*
tar -xpf $HOME/lfs-temp-tools-12.2.tar.xz

su - lfs

```

7 Installing Basic System Software

7.1 man-pages

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=man-pages
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

rm -v man3/crypt*
make prefix=/usr install
```

7.2 iana-etc

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=iana-etc
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

cp services protocols /etc
```

7.3 glibc

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=glibc
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

patch -Np1 -i ../glibc-2.40-fhs-1.patch

mkdir build
cd build

echo "rootsbindir=/usr/sbin" > configparms
../configure \
```

```

--prefix=/usr \
--disable-werror \
--enable-kernel=4.19 \
--enable-stack-protector=strong \
--disable-nscd \
libc_cv_slibdir=/usr/lib

make
make check
touch /etc/ld.so.conf
sed '/test-installation/s@$(PERL)@echo not running@' -i ../Makefile
make install
sed '/RTLDLIST=/s@/usr@@g' -i /usr/bin/ldd

localedef -i C -f UTF-8 C.UTF-8
localedef -i cs_CZ -f UTF-8 cs_CZ.UTF-8
localedef -i de_DE -f ISO-8859-1 de_DE
localedef -i de_DE@euro -f ISO-8859-15 de_DE@euro
localedef -i de_DE -f UTF-8 de_DE.UTF-8
localedef -i el_GR -f ISO-8859-7 el_GR
localedef -i en_GB -f ISO-8859-1 en_GB
localedef -i en_GB -f UTF-8 en_GB.UTF-8
localedef -i en_HK -f ISO-8859-1 en_HK
localedef -i en_PH -f ISO-8859-1 en_PH
localedef -i en_US -f ISO-8859-1 en_US
localedef -i en_US -f UTF-8 en_US.UTF-8
localedef -i es_ES -f ISO-8859-15 es_ES@euro
localedef -i es_MX -f ISO-8859-1 es_MX
localedef -i fa_IR -f UTF-8 fa_IR
localedef -i fr_FR -f ISO-8859-1 fr_FR
localedef -i fr_FR@euro -f ISO-8859-15 fr_FR@euro
localedef -i fr_FR -f UTF-8 fr_FR.UTF-8
localedef -i is_IS -f ISO-8859-1 is_IS
localedef -i is_IS -f UTF-8 is_IS.UTF-8
localedef -i it_IT -f ISO-8859-1 it_IT
localedef -i it_IT -f ISO-8859-15 it_IT@euro
localedef -i it_IT -f UTF-8 it_IT.UTF-8
localedef -i ja_JP -f EUC-JP ja_JP
localedef -i ja_JP -f SHIFT_JIS ja_JP.SJIS 2> /dev/null || true
localedef -i ja_JP -f UTF-8 ja_JP.UTF-8

```

```

localedef -i nl_NL@euro -f ISO-8859-15 nl_NL@euro
localedef -i ru_RU -f KOI8-R ru_RU.KOI8-R
localedef -i ru_RU -f UTF-8 ru_RU.UTF-8
localedef -i se_NO -f UTF-8 se_NO.UTF-8
localedef -i ta_IN -f UTF-8 ta_IN.UTF-8
localedef -i tr_TR -f UTF-8 tr_TR.UTF-8
localedef -i zh_CN -f GB18030 zh_CN.GB18030
localedef -i zh_HK -f BIG5-HKSCS zh_HK.BIG5-HKSCS
localedef -i zh_TW -f UTF-8 zh_TW.UTF-8

```

8 Configuring glibc

8.1 Adding nsswitch.conf

```

cat > /etc/nsswitch.conf << "EOF"
# Begin /etc/nsswitch.conf
passwd: files
group: files
shadow: files
hosts: files dns
networks: files
protocols: files
services: files
ethers: files
rpc: files
# End /etc/nsswitch.conf
EOF

```

8.2 Adding Time Zone Data

```

tar -xf ../../tzdata2024a.tar.gz
ZONEINFO=/usr/share/zoneinfo
mkdir -pv $ZONEINFO/{posix,right}
for tz in etcetera southamerica northamerica europe africa antarctica asia australasia
do
    zic -L /dev/null -d $ZONEINFO ${tz}
    zic -L /dev/null -d $ZONEINFO/posix ${tz}
    zic -L leapseconds -d $ZONEINFO/right ${tz}
done
cp -v zone.tab zone1970.tab iso3166.tab $ZONEINFO

```

```
zic -d $ZONEINFO -p America/New_York
unset ZONEINFO
#tzselect
ln -sfv /usr/share/zoneinfo/<xxx> /etc/localtime
```

8.3 Configuring the Dynamic Loader

```
cat > /etc/ld.so.conf << "EOF"
# Begin /etc/ld.so.conf
/usr/local/lib
/opt/lib
EOF

cat >> /etc/ld.so.conf << "EOF"
# Add an include directory
include /etc/ld.so.conf.d/*.conf
EOF
mkdir -pv /etc/ld.so.conf.d
```

8.4 zlib

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=zlib
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
make check
make install
rm -fv /usr/lib/libz.a
```

8.5 bzip

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=bzip
```



```

tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

patch -Np1 -i ../bzip2-1.0.8-install_final_docs-1.patch
sed -i 's@(\ln -s -f \)$(PREFIX)/bin/@\1@' Makefile
sed -i "s@(PREFIX)/man@(PREFIX)/share/man@g" Makefile

make -f Makefile-libbz2_so
make clean

make
make PREFIX=/usr install
cp -av libbz2.so.* /usr/lib
ln -sv libbz2.so.1.0.8 /usr/lib/libbz2.so

cp -v bzip2-shared /usr/bin/bzip2
for i in /usr/bin/{bzip2cat,bunzip2}; do
    ln -sfv bzip2 $i
done

rm -fv /usr/lib/libbz2.a

```

8.6 xz

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=xz
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/xz-5.6.2

make
make check
make install

```

8.7 lz4

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=lz4
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

make BUILD_STATIC=no PREFIX=/usr
make -j1 check
make BUILD_STATIC=no PREFIX=/usr install
```

8.8 zstd

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=zstd
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

make prefix=/usr
make check
make prefix=/usr install
rm -v /usr/lib/libzstd.a
```

8.9 file

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=file
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

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

```
make
make check
make install
```

8.10 readline

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=readline
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

sed -i 'MV.*old/d' Makefile.in
sed -i '{OLD_SUFFIX}/c:' support/shlib-install

sed -i 's/-Wl,-rpath,[^ ]*//' support/shobj-conf

./configure \
--prefix=/usr \
--disable-static \
--with-curses \
--docdir=/usr/share/doc/readline-8.2.13

make SHLIB_LIBS="-lcursesw"
make SHLIB_LIBS="-lcursesw" install
install -v -m644 doc/*.{ps,pdf,html,dvi} /usr/share/doc/readline-8.2.13
```

8.11 m4

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=m4
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

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

```
make check
make install
```

8.12 bc

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=bc
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

CC=gcc ./configure --prefix=/usr -G -O3 -r
make
make test
make install
```

8.13 flex

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=flex
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--docdir=/usr/share/doc/flex-2.6.4 \
--disable-static

make
make check
make install

ln -sv flex /usr/bin/lex
ln -sv flex.1 /usr/share/man/man1/lex.1
```

8.14 tcl

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=tcl*src
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

SRCDIR=$(pwd)
cd unix
./configure --prefix=/usr \
--mandir=/usr/share/man \
--disable-rpath

make
sed -e "s|$SRCDIR/unix|usr/lib|" \
    -e "s|$SRCDIR|usr/include|" \
    -i tclConfig.sh
sed -e "s|$SRCDIR/unix/pkgs/tdbc1.1.7|usr/lib/tdbc1.1.7|" \
    -e "s|$SRCDIR/pkgs/tdbc1.1.7/generic|usr/include|" \
    -e "s|$SRCDIR/pkgs/tdbc1.1.7/library|usr/lib/tcl8.6|" \
    -e "s|$SRCDIR/pkgs/tdbc1.1.7|usr/include|" \
    -i pkgs/tdbc1.1.7/tdbcConfig.sh
sed -e "s|$SRCDIR/unix/pkgs/itcl4.2.4|usr/lib/itcl4.2.4|" \
    -e "s|$SRCDIR/pkgs/itcl4.2.4/generic|usr/include|" \
    -e "s|$SRCDIR/pkgs/itcl4.2.4|usr/include|" \
    -i pkgs/itcl4.2.4/itclConfig.sh
unset SRCDIR

make test
make isntall
chmod -v u+w /usr/libtcl8.6.so
make install-private-headers
ln -sfv tclsh8.6 /usr/bin/tclsh
mv /usr/share/man/man3/{Thread,Tcl_Thread}.3

# FIXME
# tar -xf ../tcl8.6.14-html.tar.gz --strip-components=1
# mkdir -v -p /usr/share/doc/tcl-8.6.14
```

```
# cp -v -r ./html/* /usr/share/doc/tcl-8.6.14
```

8.15 expect

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=expect
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

python3 -c 'from pty import spawn; spawn(["echo", "ok"])'
patch -Np1 -i ../expect-5.45.4-gcc14-1.patch
./configure \
--prefix=/usr \
--with-tcl=/usr/lib \
--enable-shared \
--disable-rpath \
--mandir=/usr/share/man \
--with-tclinclude=/usr/include

make
make test
make install
# FIXME
ln -svf expect5.45.4/libexpect5.45.4.so /usr/lib
```

8.16 dejaGNU

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=dejagnu
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

mkdir build
cd build
```

```

../configure --prefix=/usr
makeinfo --html --no-split -o doc/dejagnum.html ../doc/dejagnum.texi
makeinfo --plaintext -o doc/dejagnum.txt ../doc/dejagnum.texi

make check
make install
# FIXME
install -v -dm755 /usr/share/doc/dejagnum-1.6.3
install -v -m644 doc/dejagnum.{html,txt} /usr/share/doc/dejagnum-1.6.3

cd ..

```

8.17 pkgconf

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=pkgconf
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure --prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/pkgconf-2.3.0

make
make install

ln -sv pkgconf /usr/bin/pkg-config
ln -sv pkgconf.1 /usr/share/man/man1/pkg-config.1

```

8.18 binutils

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=binutils
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

```

```

mkdir build
cd build

../configure \
--prefix=/usr \
--sysconfdir=/etc \
--enable-gold \
--enable-ld=default \
--enable-plugins \
--enable-shared \
--disable-werror \
--enable-64-bit-bfd \
--enable-new-dtags \
--with-system-zlib \
--enable-default-hash-style=gnu

make tooldir=/usr

# FIXME: DO NOT SKIP
make -k check
grep '^FAIL:' $(find -name '*.log')

make tooldir=/usr install
rm -fv /usr/lib/lib{bfd,ctf,ctf-nobfd,gprofng,opcodes,sframe}.a

cd ../../

```

8.19 gmp

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=gmp
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--enable-cxx \

```



```

--disable-static \
--docdir=/usr/share/doc/gmp-6.3.0

make
make html

# FIXME: DO NOT SKIP
make check 2>&1 | tee gmp-check-log
awk '/# PASS:/{total+= $3} ; END{print total}' gmp-check-log

make install
make install-html

```

8.20 mpfr

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=mpfr
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--enable-thread-safe \
--docdir=/usr/share/doc/mpfr-4.2.1

make
make html

# FIXME: DO NOT SKIP
make check

make install
make install-html

```

8.21 mpc

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=mpc
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/mpc-1.3.1

make
make html

make check

make install
make install-html
```

8.22 attr

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=attr
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--sysconfdir=/etc \
--docdir=/usr/share/doc/attr-2.5.2

make
make check
make install
```

8.23 acl

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=acl
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/acl-2.3.2

make
make install
```

8.24 libcap

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=libcap
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

sed -i '/install -m.*STA/d' libcap/Makefile
make prefix=/usr lib=lib

make test
make prefix=/usr lib=lib install
```

8.25 libxcrypt

```
#!/bin/sh

cd $LFS/sources
```

```

LFS_TARGET=libxcrypt
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--enable-hashes=strong,glibc \
--enable-obsolete-api=no \
--disable-static \
--disable-failure-tokens

make
make check
make install

```

8.26 shadow

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=shadow
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

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 / /' {} \;

touch /usr/bin/passwd
./configure \
--sysconfdir=/etc \
--disable-static \
--with-{b,yes}crypt \
--without-libbsd \
--with-group-name-max-length=32

make
make exec_prefix=/usr install

```

```
make -C man install-man
```

```
# configuring  
pwconv  
grpconv
```

```
mkdir -p /etc/default  
useradd -D --gid 999
```

```
passwd root
```

8.27 gcc

```
#!/bin/sh
```

```
cd $LFS/sources  
LFS_TARGET=gcc  
tar -xf $LFS_TARGET.tar*  
cd $LFS_TARGET*/
```

```
case $(uname -m) in  
    x86_64)  
        sed -e '/m64=/s/lib64/lib/' \  
            -i.orig gcc/config/i386/t-linux64  
        ;;  
esac
```

```
mkdir build  
cd build
```

```
../configure \  
--prefix=/usr \  
LD=ld \  
--enable-languages=c,c++ \  
--enable-default-pie \  
--enable-default-ssp \  
--enable-host-pie \  
--disable-multilib \  
--disable-bootstrap \  

```

```

--disable-fixincludes \
--with-system-zlib

make

# stack limit
ulimit -s -H unlimited

# 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

chown -R tester .
su tester -c "PATH=$PATH make -k check"
../contrib/test_summary

make install

# ownership to root user and group
chown -v -R root:root \
    /usr/lib/gcc/$(gcc -dumpmachine)/14.2.0/include{,-fixed}

# fhs historical reasons
ln -svr /usr/bin/cpp /usr/lib

#man page symlink
ln -sv gcc.1 /usr/share/man/man1/cc.1

# sanity checks FIXME
echo 'int main(){}' > dummy.c
cc dummy.c -v -Wl,--verbose &> dummy.log
readelf -l a.out | grep ': /lib'

# correct header files FIXME
grep -E -o '/usr/lib.*S?crt[1in].*succeeded' dummy.log
# linker used with correct search paths FIXME

```

```

grep -B4 '^ /usr/include' dummy.log
# correct libc FIXME
grep 'SEARCH.*/usr/lib' dummy.log |sed 's|; |\n|g'
# correct dinamic linker FIXME
grep found dummy.log

# rm -v dummy.c a.out dummy.log

mkdir -pv /usr/share/gdb/auto-load/usr/lib
mv -v /usr/lib/*gdb.py /usr/share/gdb/auto-load/usr/lib

cd ../../

```

8.28 ncurses

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=ncurses
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--mandir=/usr/share/man \
--with-shared \
--without-debug \
--without-normal \
--with-cxx-shared \
--enable-pc-files \
--with-pkg-config-libdir=/usr/lib/pkgconfig

make

make DESTDIR=$PWD/dest install
install -vm755 dest/usr/lib/libncursesw.so.6.5 /usr/lib
rm -v dest/usr/lib/libncursesw.so.6.5
sed -e 's/^#if.*XOPEN.*$/#if 1/' \
-i dest/usr/include/curses.h

```

```

cp -av dest/* /

for lib in ncurses form panel menu ; do
    ln -sfv lib${lib}w.so /usr/lib/lib${lib}.so
    ln -sfv ${lib}w.pc /usr/lib/pkgconfig/${lib}.pc
done

ln -sfv libncursesw.so /usr/lib/libcurses.so
cp -v -R doc -T /usr/share/doc/ncurses-6.5

```

8.29 sed

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=sed
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

cp -v -R doc -T /usr/share/doc/ncurses-6.5
make
make html

chown -R tester .
su tester -c "PATH=$PATH make check"

make install
install -d -m755 /usr/share/doc/sed-4.9
install -m644 doc/sed.html /usr/share/doc/sed-4.9

```

8.30 psmisc

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=psmisc
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

```



```
./configure --prefix=/usr
make
make check
make install
```

8.31 gettext

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=gettext
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/gettext-0.22.5

make
make check
make install
chmod -v 0755 /usr/preloadable_libnsl.so
```

8.32 bison

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=bison
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr --docdir=/usr/share/doc/bison-3.8.2
make
make check
make install
```

8.33 grep

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=grep
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

sed -i "s/echo/#echo/" src/egrep.sh
./configure --prefix=/usr
make
make check
make install
```

8.34 bash

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=bash
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--without-bash-malloc \
--with-installed-readline \
bash_cv_strtold_broken=no \
--docdir=/usr/share/doc/bash-5.2.32

make
chown -R tester .

su -s /usr/bin/expect tester << "EOF"
set timeout -1
spawn make tests
expect eof
lassign [wait] _ _ _ value
exit $value
```

EOF

make install

exec /usr/bin/bash --login

8.35 libtool

```
#!/bin/sh
```

```
cd $LFS/sources
LFS_TARGET=libtool
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/
```

```
./configure --prefix=/usr
make
make -k check
make install
rm -fv /usr/lib/libltdl.a
```

8.36 gdbm

```
#!/bin/sh
```

```
cd $LFS/sources
LFS_TARGET=gdbm
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/
```

```
./configure \
--prefix=/usr \
--disable-static \
--enable-libgdbm-compat
```

```
make
make check
make install
```

8.37 gperf

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=gperf
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure --prefix=/usr --docdir=/usr/share/doc/gperf-3.1
make
make -j1 check
make install
```

8.38 expat

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=expat
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--docdir=/usr/share/doc/expat-2.6.2

make
make check
make install

#FIXME
install -v -m644 doc/*.html,css /usr/share/doc/expat-2.6.2
```

8.39 inetutils

```
#!/bin/sh
```

```

cd $LFS/sources
LFS_TARGET=inetutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

sed -i 's/def HAVE_TERMCAP_TGETENT/ 1/' telnet/telnet.c

./configure \
--prefix=/usr \
--bindir=/usr/bin \
--localstatedir=/var \
--disable-logger \
--disable-whois \
--disable-rcp \
--disable-rexec \
--disable-rlogin \
--disable-rsh \
--disable-servers

make
make check
make install
mk -v /usr/{,s}bin/ifconfig

```

8.40 less

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=less
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr --sysconfdir=/etc
make
make check
make install

```

8.41 perl

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=perl
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

export BUILD_ZLIB=False
export BUILD_BZIP2=0

sh Configure -des \
-D prefix=/usr \
-D vendorprefix=/usr \
-D privlib=/usr/lib/perl5/5.40/core_perl \
-D archlib=/usr/lib/perl5/5.40/core_perl \
-D sitelib=/usr/lib/perl5/5.40/site_perl \
-D sitearch=/usr/lib/perl5/5.40/site_perl \
-D vendorlib=/usr/lib/perl5/5.40/vendor_perl \
-D vendorarch=/usr/lib/perl5/5.40/vendor_perl \
-D man1dir=/usr/share/man/man1 \
-D man3dir=/usr/share/man/man3 \
-D pager="/usr/bin/less -isR" \
-D useshrplib \
-D usethreads

make
TEST_JOBS=$(nproc) make test_harness

make install
unset BUILD_ZLIB BUILD_BZIP2
```

8.42 xml::parser

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=XML-Parser
tar -xf $LFS_TARGET*tar*
```

```
cd $LFS_TARGET*/
```

```
perl Makefile.PL  
make  
make test  
make install
```

8.43 intltool

```
#!/bin/sh
```

```
cd $LFS/sources  
LFS_TARGET=intltool  
tar -xf $LFS_TARGET*tar*  
cd $LFS_TARGET*/
```

```
sed -i 's:\\\\${:\\\\$\\{:intltool-update.in  
./configure --prefix=/usr  
make  
make check  
make install  
install -v -Dm644 doc/I18N-HOWTO /usr/share/doc/intltool-0.51.0/I18N-HOWTO
```

8.44 autoconf

```
#!/bin/sh
```

```
cd $LFS/sources  
LFS_TARGET=autoconf  
tar -xf $LFS_TARGET*tar*  
cd $LFS_TARGET*/
```

```
./configure --prefix=/usr  
make  
make check  
make install
```

8.45 automake

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=automake
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

#FIXME
./configure --prefix=/usr --docdir=/usr/share/doc/automake-1.17
make
make -j$(($(nproc)>4?$(nproc):4)) check
make install
```

8.46 openssl

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=openssl
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./config \
--prefix=/usr \
--openssldir=/etc/ssl \
--libdir=lib \
shared \
zlib-dynamic

make
HARNESS_JOBS=$(nproc) make test

sed -i 'INSTALL_FINAL_LIBS/s/libcrypto.a libssl.a/' Makefile
make MANSUFFIX=ssl install

mv -v /usr/share/doc/openssl /usr/share/doc/openssl-3.3.1

#FIXME
```



```
cp -vfr doc/* /usr/share/doc/openssl-3.3.1
```

8.47 kmod

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=kmod
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--sysconfdir=/etc \
--with-openssl \
--with-xz \
--with-zstd \
--with-zlib \
--disable-manpages

make
make install
for target in depmod insmod modinfo modprobe rmmod; do
    ln -sfv ../bin/kmod /usr/sbin/$target
    rm -fv /usr/bin/$target
done
```

8.48 libelf

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=elfutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

cd libelf/

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

```

--disable-debuginfod \
--enable-libdebuginfod=dummy

make
make check

make -C libelf install
install -vm644 config/libelf.pc /usr/lib/pkgconfig
rm /usr/lib/libelf.a

cd ../../

```

8.49 libffi

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=libffi
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \
--prefix=/usr \
--disable-static \
--with-gcc-arch=native

make
make check
make install

```

8.50 python

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=python
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure \

```

```

--prefix=/usr \
--enable-shared \
--with-system-expat \
--enable-optimizations

make
make test TESTOPTS="--timeout 120"
make install

# FIXME
install -v -dm755 /usr/share/doc/python-3.12.5/html
tar --no-same-owner \
    -xvf ../python-3.12.5-docs-html.tar.bz2
cp -R --no-preserve=mode python-3.12.5-docs-html/* \
    /usr/share/doc/python-3.12.5/html

```

8.51 flit_{core}

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=flit_core
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

pip3 wheel -w dist --no-cache-dir --no-build-isolation --no-deps $PWD
pip3 install --no-index --find-links=dist wheel

```

8.52 setuptools

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=setuptools
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

pip3 wheel -w dist --no-cache-dir --no-build-isolation --no-deps $PWD
pip3 install --no-index --find-links=dist setuptools

```

8.53 ninja

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=ninja
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

export NINJAJOBS=4

sed -i '/int Guess/a \
    int j = 0;\
    char* jobs = getenv( "NINJAJOBS" );\
    if ( jobs != NULL ) j = atoi( jobs );\
    if ( j > 0 ) return j;\
' src/ninja.cc

python3 configure.py --bootstrap

install -vm755 ninja /usr/bin/
install -vDm644 misc/bash-completion /usr/share/bash-completion/completions/ninja
install -vDm644 misc/zsh-completion /usr/share/zsh/site-functions/_ninja
```

8.54 meson

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=meson
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

pip3 wheel -w dist --no-cache-dir --no-build-isolation --no-deps $PWD

pip3 install --no-index --find-links dist meson
install -vDm644 data/shell-completions/bash/meson /usr/share/bash-completion/completions/meson
install -vDm644 data/shell-completions/zsh/_meson /usr/share/zsh/site-functions/_meson
```

8.55 coreutils

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=coreutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

patch -Np1 -i ../coreutils-9.5-i18n-2.patch

autoreconf -fiv
FORCE_UNSAFE_CONFIGURE=1 ./configure \
--prefix=/usr \
--enable-no-install-program=kill,uptime

make
make NON_ROOT_USERNAME=tester check-root
groupadd -g 102 dummy -U tester
chown -R tester .

su tester -c "PATH make -k RUN_EXPENSIVE_TESTS=yes check" \
    < /dev/null

groupdel dummy
make install

mv -v /usr/bin/chroot /usr/sbin
mv -v /usr/share/man/man1/chroot.1 /usr/share/man/man8/chroot.8
sed -i 's/"1"/"8"/' /usr/share/man/man8/chroot.8
```

8.56 check

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=check
tar -xf $LFS_TARGET*tar*
```

```

cd $LFS_TARGET*/

../configure --prefix=/usr --disable-static
make
make check
make docdir=/usr/share/doc/check-0.15.2 install

```

8.57 diffutils

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=diffutils
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
make check
make install

```

8.58 gawk

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=gawk
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

sed -i 's/extras//' Makefile.in
./configure --prefix=/usr
make

chown -R tester .
su tester -c "PATH=$PATH make check"

rm -f /usr/bin/gawk-5.3.0
make install

```

```
ln -sv gawk.1 /usr/share/man/man1/awk.1

#FIXME
mkdir -pv /usr/share/doc/gawk-5.3.0
cp -v doc/{awkforai.txt,*.{eps,pdf,jpg}} /usr/share/doc/gawk-5.3.0
```

8.59 findutils

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=findutils
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure --prefix=/usr --localstatedir=/var/lib/locate
make

chown -R tester .
su tester -c "PATH=$PATH make check"

make install
```

8.60 groff

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=groff
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

PAGE=<paper_size> ./configure --prefix=/usr
make
make check
make install
```

8.61 grub

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=grub
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

unset {C,CPP,CXX,LD}FLAGS

echo depends bli part_gpt > grub-core/extra_deps.lst

./configure \
--prefix=/usr \
--sysconfdir=/etc \
--disable-efiemu \
--disable-werror

make
make install
mv -v /etc/bash_completion.d/grub /usr/share/bash-completion/completions
```

8.62 gzip

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=gzip
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
make check
make install
```

8.63 iproute

```
#!/bin/sh
```



```

cd $LFS/sources
LFS_TARGET=iproute
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

sed -i /ARPD/d Makefile
rm -fv man/man8/arpd.8

make NETNS_RUN_DIR=/run/netns
make SBINDIR=/usr/sbin install

#FIXME
mkdir -pv /usr/share/doc/iproute2-6.10.0
cp -v COPYING README* /usr/share/doc/iproute2-6.10.0

```

8.64 kbd

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=kbd
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

patch -Np1 -i ../kbd-2.6.4-backspace-1.patch
sed -i '/RESIZECONS_PROGS=/s/yes/no/' configure
sed -i 's/resizecons.8 //' docs/man/man8/Makefile.in
./configure --prefix=/usr --disable-vlock
make
make check
make install

#FIXME
cp -R -v docs/doc -T /usr/share/doc/kbd-2.6.4

```

8.65 libpipeline

```

#!/bin/sh

```

```

cd $LFS/sources
LFS_TARGET=libpipeline
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

cp -R -v docs/doc -T /usr/share/doc/kbd-2.6.4
make
make check
make install

```

8.66 make

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=make
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
chown -R tester .
su tester -c "PATH=$PATH make check"
make install

```

8.67 patch

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=patch
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
make check
make install

```

8.68 tar

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=tar
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

FORCE_UNSAFE_CONFIGURE=1 \
    ./configure --prefix=/usr
make
make check

make install
make -C doc install-html docdir=/usr/share/doc/tar-1.35
```

8.69 texinfo

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=texinfo
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

./configure --prefix=/usr
make
make check
make install
make TEXMF=/usr/share/texmf install-tex

pushd /usr/share/info
    rm -v dir
    for f in *
        do install-info $f dir 2>/dev/null
    done
popd
```

8.70 vim

```
#!/bin/sh

cd $LFS/sources
LFS_TARGET=vim
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

echo '#define SYS_VIMRC_FILE "/etc/vimrc"' >> src/feature.h
./configure --prefix=/usr
make
chown -R tester .
# FIXME
su tester -c "TERM=xterm-256color LANG=en_US-8 make -j1 test" \
    &> vim-test.log

make install
ln -sv vim /usr/bin/vi
for L in /usr/share/man/{,*/}man/vim.1; do
    ln -sv vim.1 $(dirname $L)/vi.1
done

ln -sv ../vim/vim91/doc /usr/share/doc/vim-9.1.0660

# Configuring Vim
cat > /etc/vimrc << "EOF"
" Begin /etc/vimrc
" Ensure defaults are set before customizing settings, not after
source $VIMRUNTIME/defaults.vim
let skip_defaults_vim=1
set nocompatible
set backspace=2
set mouse=
syntax on
if (&term == "xterm") || (&term == "putty")
set background=dark
endif
" End /etc/vimrc
EOF
```

```
cd ..
```

```
#set spelllang=en,ru  
#set spell
```

8.71 MarkupSafe

```
#!/bin/sh
```

```
cd $LFS/sources  
LFS_TARGET=MarkupSafe  
tar -xf $LFS_TARGET*tar*  
cd $LFS_TARGET*/
```

```
pip3 wheel -w dist --no-cache-dir --no-build-isolation --no-deps $PWD  
pip3 install --no-index --no-user --find-links dist Markupsafe
```

8.72 jinja

```
#!/bin/sh
```

```
cd $LFS/sources  
LFS_TARGET=jinja  
tar -xf $LFS_TARGET*tar*  
cd $LFS_TARGET*/
```

```
pip3 wheel -w dist --no-cache-dir --no-build-isolation --no-deps $PWD  
pip3 install --no-index --no-user --find-links dist Jinja2
```

8.73 udev

```
#!/bin/sh
```

```
cd $LFS/sources  
LFS_TARGET=systemd  
tar -xf $LFS_TARGET*tar*  
cd $LFS_TARGET*/
```

```

sed -i -e 's/GROUP="render"/GROUP="video"/' \
    -e 's/GROUP="sgx", //' rules.d/50-udev-default.rules.in

sed '/systemd-sysctl/s/^/#/' -i rules.d/99-systemd.rules.in
sed '/NETWORK_DIRS/s/systemd/udev/' -i src/basic/path-lookup.h

mkdir build
cd build

meson setup .. \
--prefix=/usr \
--buildtype=release \
-D mode=release \
-D dev-kvm-mode=0660 \
-D link-udev-shared=false \
-D logind=false \
-D vconsole=false

export udev_helpers=$(grep "'name' :" ../src/udev/meson.build | \
    awk '{print $3}' | tr -d ",'" | grep -v 'udevadm')

ninja udevadm systemd-hwdb \
$(ninja -n | grep -Eo '(src/(lib)?udev|rules.d|hwdb.d)/[^\ ]*') \
$(realpath libudev.so --relative-to .) \
$udev_helpers

install -vm755 -d {/usr/lib,/etc}/udev/{hwdb.d,rules.d,network}
install -vm755 -d /usr/{lib,share}/pkgconfig
install -vm755 udevadm /usr/bin/
install -vm755 systemd-hwdb /usr/bin/udev-hwdb
ln -svfn ../bin/udevadm /usr/sbin/udev
cp -av libudev.so{,[0-9]} /usr/lib/
install -vm644 ../src/libudev/libudev.h /usr/include/
install -vm644 src/libudev/*.pc /usr/lib/pkgconfig/
install -vm644 src/udev/*.pc /usr/share/pkgconfig/
install -vm644 ../src/udev/udev.conf /etc/udev/
install -vm644 rules.d/* ../rules.d/README /usr/lib/udev/rules.d/
install -vm644 $(find ../rules.d/*.rules \
    -not -name '*power-switch*') /usr/lib/udev/rules.d/
install -vm644 hwdb.d/* ../hwdb.d/{*.hwdb,README} /usr/lib/udev/hwdb.d/

```

```

install -vm755 $udev_helpers /usr/lib/udev
install -vm644 ../network/99-default.link /usr/lib/udev/network

tar -xvf ../../udev-lfs-20230818.tar.xz
make -f udev-lfs-20230818/Makefile.lfs install

#FIXME no man pages

unset udev_helpers

udev-hwdb update

cd ../../

```

8.74 procps-ng

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=procps-ng
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

./configure --prefix=/usr \
--docdir=/usr/share/doc/procps-ng-4.0.4 \
--disable-static \
--disable-kill

make
chown -R tester .
su tester -c "PATH=$PATH make check"
make install

```

8.75 util-linux

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=util-linux
tar -xf $LFS_TARGET.tar*

```

```

cd $LFS_TARGET*/

./configure \
--bindir=/usr/bin \
--libdir=/usr/lib \
--runstatedir=/run \
--sbindir=/usr/sbin \
--disable-chfn-chsh \
--disable-login \
--disable-nologin \
--disable-su \
--disable-setpriv \
--disable-runuser \
--disable-pylibmount \
--disable-liblastlog2 \
--disable-static \
--without-python \
--without-systemd \
--without-systemdsystemunitdir\
ADJTIME_PATH=/var/lib/hwclock/adjtime \
--docdir=/usr/share/doc/util-linux-2.40.2

make

touch /etc/fstab
chown -R tester .
su tester -c "make -k check"

make install

```

8.76 e2fsprogs

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=e2fsprogs
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

```



```

mkdir build
cd build

../configure \
--prefix=/usr \
--sysconfdir=/etc \
--enable-elf-shlibs \
--disable-libblkid \
--disable-libuuid \
--disable-uidd \
--disable-fsck

make
make check
make install

rm -fv /usr/lib/{libcom_err,libe2p,libext2fs,libss}.a

gunzip -v /usr/share/info/libext2fs.info.gz
install-info --dir-file=/usr/share/info/dir /usr/share/info/libext2fs.info

#FIXME docs
makeinfo -o doc/com_err.info ../lib/et/com_err.texinfo
install -v -m644 doc/com_err.info /usr/share/info
install-info --dir-file=/usr/share/info/dir /usr/share/info/com_err.info

# configuring
sed 's/metadata_csum_seed,/' -i /etc/mke2fs.conf

```

8.77 sysklogd

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=sysklogd
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

../configure \

```

```

--prefix=/usr \
--sysconfdir=/etc \
--runstatedir=/run \
--without-logger

make
make install

#configuring
cat > /etc/syslog.conf << "EOF"
# Begin /etc/syslog.conf
auth,authpriv.* -/var/log/auth.log
*.*;auth,authpriv.none -/var/log/sys.log
daemon.* -/var/log/daemon.log
kern.* -/var/log/kern.log
mail.* -/var/log/mail.log
user.* -/var/log/user.log
*.emerg *
# Do not open any internet ports.
secure_mode 2
# End /etc/syslog.conf
EOF

```

8.78 sysvinit

```

#!/bin/sh

cd $LFS/sources
LFS_TARGET=sysvinit
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

patch -Np1 -i ../sysvinit-3.10-consolidated-1.patch
make
make install

```

8.79 stripping skipped FIXME

9 System Configuration

9.1 lfs-bootscripts

```
cd $LFS/sources
LFS_TARGET=lfs-bootscripts
tar -xf $LFS_TARGET*tar*
cd $LFS_TARGET*/

make install
```

9.2 Creating Custom Udev Rules

```
bash /usr/lib/udev/init-net-rules.sh
cat /etc/udev/rules.d/70-persistent-net.rules

#sed -e '/^AlternativeNamesPolicy/s/=.*$/=/' \
#/usr/lib/udev/network/99-default.link \
#> /etc/udev/network/99-default.link
```

9.3 CD_{ROM} Symlinks

```
udevadm test /sys/block/hdd

#sed -e 's/"write_cd_rules"/"write_cd_rules mode"/' \
# -i /etc/udev/rules.d/83-cdrom-symlinks.rules
```

9.4 Duplicate devices

10 Network Configuration

10.1 Creating Network Interface configuration files

```
cd /etc/sysconfig/
cat > ifconfig.eth0 << "EOF"
ONBOOT=yes
IFACE=eth0
SERVICE=ipv4-static
IP=192.168.1.2
```

```
GATEWAY=192.168.1.1
PREFIX=24
BROADCAST=192.168.1.255
EOF
```

10.2 Creating the /etc/resolv.conf

```
cat > /etc/resolv.conf << "EOF"
# Begin /etc/resolv.conf
domain <Your Domain Name>
nameserver <IP address of your primary nameserver>
nameserver <IP address of your secondary nameserver>
# End /etc/resolv.conf
EOF
```

10.3 configuring the system hostname

```
echo "<lfs>" > /etc/hostname
```

10.4 Customizing the /etc/hosts

```
cat > /etc/hosts << "EOF"
# Begin /etc/hosts
127.0.0.1 localhost.localdomain localhost
127.0.1.1 <FQDN> <HOSTNAME>
<192.168.1.1> <FQDN> <HOSTNAME> [alias1] [alias2 ...]
::1
    localhost ip6-localhost ip6-loopback
ff02::1
    ip6-allnodes
ff02::2
    ip6-allrouters
# End /etc/hosts
EOF
```

11 System V Bootscript Usage and Configuration

11.1 Configuring SysVinit

```
cat > /etc/inittab << "EOF"
# Begin /etc/inittab
```

```

id:3:initdefault:
si::sysinit:/etc/rc.d/init.d/rc S
l0:0:wait:/etc/rc.d/init.d/rc 0
l1:S1:wait:/etc/rc.d/init.d/rc 1
l2:2:wait:/etc/rc.d/init.d/rc 2
l3:3:wait:/etc/rc.d/init.d/rc 3
l4:4:wait:/etc/rc.d/init.d/rc 4
l5:5:wait:/etc/rc.d/init.d/rc 5
l6:6:wait:/etc/rc.d/init.d/rc 6
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now
su:S06:once:/sbin/sulogin
s1:1:respawn:/sbin/sulogin
1:2345:respawn:/sbin/agetty --noclear tty1 9600
2:2345:respawn:/sbin/agetty tty2 9600
3:2345:respawn:/sbin/agetty tty3 9600
4:2345:respawn:/sbin/agetty tty4 9600
5:2345:respawn:/sbin/agetty tty5 9600
6:2345:respawn:/sbin/agetty tty6 9600
# End /etc/inittab
EOF

```

11.2 Configuring the System Clock

```

cat > /etc/sysconfig/clock << "EOF"
# Begin /etc/sysconfig/clock
UTC=1
# Set this to any options you might need to give to hwclock,
# such as machine hardware clock type for Alphas.
CLOCKPARAMS=
# End /etc/sysconfig/clock
EOF

```

11.3 Linux Console

```

cat > /etc/sysconfig/console << "EOF"
# Begin /etc/sysconfig/console
UNICODE="1"
FONT="Lat2-Terminus16"
# End /etc/sysconfig/console
EOF

```

11.4 rc.site

```
# rc.site
# Optional parameters for boot scripts.
# Distro Information
# These values, if specified here, override the defaults
#DISTRO="Linux From Scratch" # The distro name
#DISTRO_CONTACT="lfs-dev@lists.linuxfromscratch.org" # Bug report address
#DISTRO_MINI="LFS" # Short name used in filenames for distro config
# Define custom colors used in messages printed to the screen
# Please consult 'man console_codes' for more information
# under the "ECMA-48 Set Graphics Rendition" section
#
254
Linux From Scratch - Version 12.2
# Warning: when switching from a 8bit to a 9bit font,
# the linux console will reinterpret the bold (1;) to
# the top 256 glyphs of the 9bit font. This does
# not affect framebuffer consoles
# These values, if specified here, override the defaults
#BRACKET="\033[1;34m" # Blue
#FAILURE="\033[1;31m" # Red
#INFO="\033[1;36m"
# Cyan
#NORMAL="\033[0;39m" # Grey
#SUCCESS="\033[1;32m" # Green
#WARNING="\033[1;33m" # Yellow
# Use a colored prefix
# These values, if specified here, override the defaults
#BMPREFIX=""
"
#SUCCESS_PREFIX="${SUCCESS} * ${NORMAL} "
#FAILURE_PREFIX="${FAILURE}*****${NORMAL} "
#WARNING_PREFIX="${WARNING} *** ${NORMAL} "
# Manually set the right edge of message output (characters)
# Useful when resetting console font during boot to override
# automatic screen width detection
#COLUMNS=120
# Interactive startup
#IPROMPT="yes" # Whether to display the interactive boot prompt
```

```

#itime="3"
# The amount of time (in seconds) to display the prompt
# The total length of the distro welcome string, without escape codes
#wlen=$(echo "Welcome to ${DISTRO}" | wc -c )
#welcome_message="Welcome to ${INFO}${DISTRO}${NORMAL}"
# The total length of the interactive string, without escape codes
#ilen=$(echo "Press 'I' to enter interactive startup" | wc -c )
#i_message="Press '${FAILURE}I${NORMAL}' to enter interactive startup"
# Set scripts to skip the file system check on reboot
#FASTBOOT=yes
# Skip reading from the console
#HEADLESS=yes
# Write out fsck progress if yes
#VERBOSE_FSCK=no
# Speed up boot without waiting for settle in udev
#OMIT_UDEV_SETTLE=y
# Speed up boot without waiting for settle in udev_retry
#OMIT_UDEV_RETRY_SETTLE=yes
# Skip cleaning /tmp if yes
#SKIPTMPCLEAN=no
# For setclock
#UTC=1
#CLOCKPARAMS=
# For consolelog (Note that the default, 7=debug, is noisy)
#LOGLEVEL=7
# For network
255
#HOSTNAME=mylfs
# Delay between TERM and KILL signals at shutdown
#KILLDELAY=3
# Optional sysklogd parameters
#SYSKLOGD_PARMS="-m 0"
# Console parameters
#UNICODE=1
#KEYMAP="de-latin1"
#KEYMAP_CORRECTIONS="euro2"
#FONT="lat0-16 -m 8859-15"
#LEGACY_CHARSET=

```

11.5 System locale

```
LFS_LOCALE_NAME=en_US.utf8
LC_ALL=$LFS_LOCALE_NAME locale language
LC_ALL=$LFS_LOCALE_NAME locale charmap
LC_ALL=$LFS_LOCALE_NAME locale int_curr_symbol
LC_ALL=$LFS_LOCALE_NAME locale int_prefix
```

```
cat > /etc/profile << "EOF"
# Begin /etc/profile
for i in $(locale); do
unset ${i%=*}
done
if [[ "$TERM" = linux ]]; then
export LANG=C.UTF-8
else
export LANG=<ll>_<CC>.<charmap><@modifiers>
fi
# End /etc/profile
EOF
```

11.6 /etc/inputrc

```
cat > /etc/inputrc << "EOF"
# Begin /etc/inputrc
# Modified by Chris Lynn <roryo@roryo.dynup.net>
# Allow the command prompt to wrap to the next line
set horizontal-scroll-mode Off
# Enable 8-bit input
set meta-flag On
set input-meta On
# Turns off 8th bit stripping
set convert-meta Off
# Keep the 8th bit for display
set output-meta On
# none, visible or audible
set bell-style none
# All of the following map the escape sequence of the value
# contained in the 1st argument to the readline specific functions
"\eOd": backward-word
```



```

"\e0c": forward-word
# for linux console
"\e[1~": beginning-of-line
"\e[4~": end-of-line
"\e[5~": beginning-of-history
"\e[6~": end-of-history
"\e[3~": delete-char
"\e[2~": quoted-insert
# for xterm
"\eOH": beginning-of-line
"\eOF": end-of-line
# for Konsole
"\e[H": beginning-of-line
"\e[F": end-of-line
# End /etc/inputrc
EOF

```

11.7 /etc/shells

```

cat > /etc/shells << "EOF"
# Begin /etc/shells
/bin/sh
/bin/bash
# End /etc/shells
EOF

```

12 Making the LFS System Bootable

12.1 Creating the /etc/fstab file

- FIXME boot

```

cat > /etc/fstab << "EOF"
# Begin /etc/fstab
# file system  mount-point  type      options      dump    fsck order
/dev/<xxx>      /              <fff>     defaults     1        1
/dev/<yyy>      swap          swap      pri=1         0        0
proc           /proc         proc      nosuid,noexec,nodev 0        0
sysfs          /sys          sysfs     nosuid,noexec,nodev 0        0
devpts         /dev/pts      devpts    gid=5,mode=620 0        0

```

```

tmpfs          /run          tmpfs          defaults          0          0
devtmpfs       /dev           devtmpfs       mode=0755,nosuid  0          0
tmpfs          /dev/shm       tmpfs          nosuid,nodev      0          0
cgroup2        /sys/fs/cgroup cgroup2        nosuid,noexec,nodev 0          0
# End /etc/fstab
EOF

```

12.2 linux

```

cd $LFS/sources
LFS_TARGET=linux
tar -xf $LFS_TARGET.tar*
cd $LFS_TARGET*/

make mrproper
make menuconfig

make
make modules_install

mount /boot

cp -iv arch/x86/boot/bzImage /boot/vmlinuz-6.10.5-lfs-12.2
cp -iv System.map /boot/System.map-6.10.5
cp -iv .config /boot/config-6.10.5
cp -r Documentation -T /usr/share/doc/linux-6.10.5

```

12.3 Configuring Linux Module load order

- /etc/modprobe.d/usb.conf

```

install -v -m755 -d /etc/modprobe.d
cat > /etc/modprobe.d/usb.conf << "EOF"
# Begin /etc/modprobe.d/usb.conf
install ohci_hcd /sbin/modprobe ehci_hcd ; /sbin/modprobe -i ohci_hcd ; true
install uhci_hcd /sbin/modprobe ehci_hcd ; /sbin/modprobe -i uhci_hcd ; true
# End /etc/modprobe.d/usb.conf
EOF

```

12.4 GRUB

TODO:

- mount boot(*dev/sdaX*) as *smth* but not boot
- move /boot/grub to smth
- move vmlinuz-linux to smth
- remount smth as boot/

```
grub-install /dev/sdb --target i386-pc
```

```
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)
```

```
menuentry "GNU/Linux, Linux 6.10.5-lfs-12.2" {
    linux /boot/vmlinuz-6.10.5-lfs-12.2 root=/dev/sda2 ro
}
EOF
```

13 USEFUL

```
ip link
ls /sys/class/net
```

```
tzselect
gcc -dumpmachine
readelf -l <name of binary> | grep interpreter
```

13.1

```
cd $LFS/sources
LFS_TARGET=
tar -xf $LFS_TARGET.tar*
```

```
cd $LFS_TARGET*/
```