

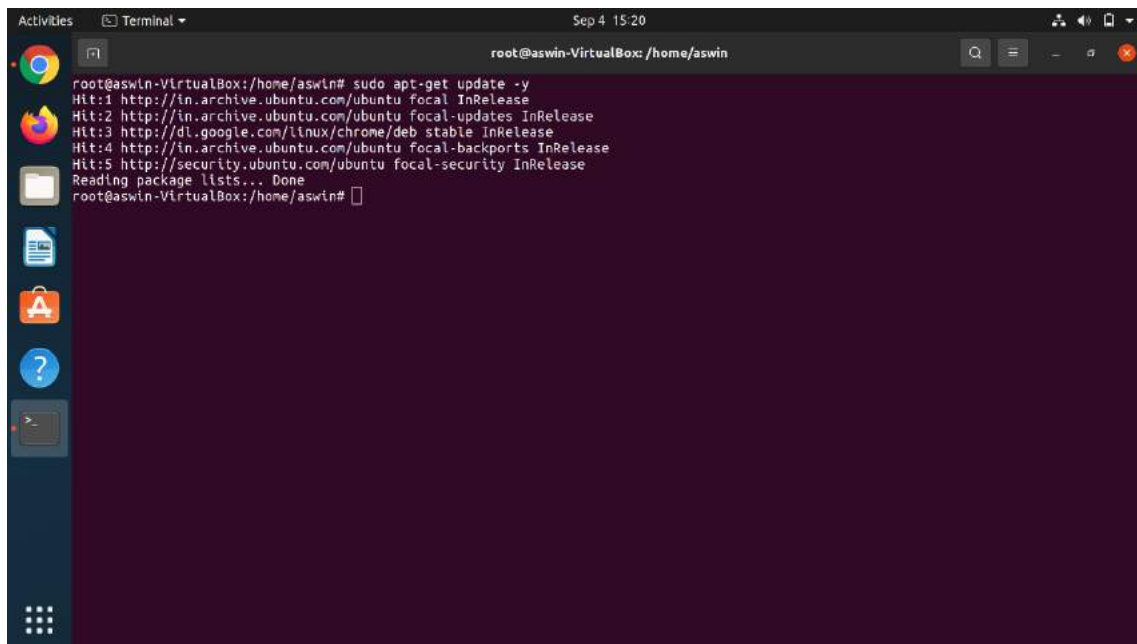
EXPIRIMENT-7:

Aim: Build and install software from source code, familiarity with make and cmake utilities expected.

Solution :-

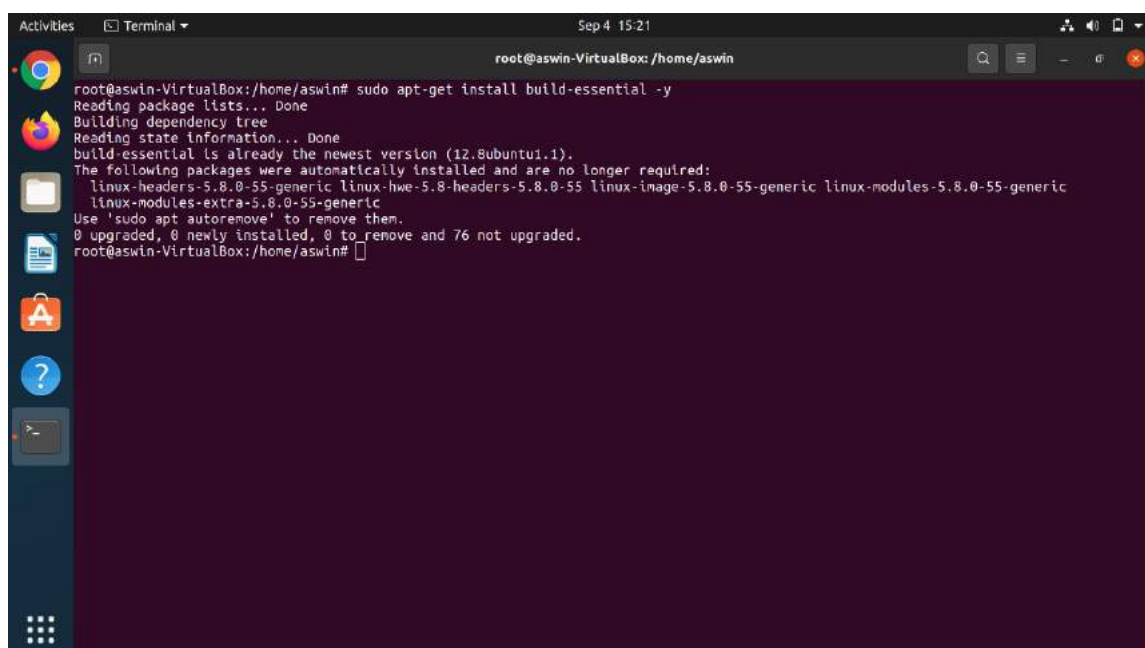
Step 1: Get the Server Ready

- As a best practice, make sure your packages are up to date:

A terminal window titled 'root@aswin-VirtualBox: /home/aswin' showing the output of the command 'sudo apt-get update -y'. The output lists five updates from various sources (in.archive.ubuntu.com, dl.google.com, security.ubuntu.com) and confirms that package lists are done.

```
root@aswin-VirtualBox: /home/aswin# sudo apt-get update -y
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://dl.google.com/linux/chrome/deb stable InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:5 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
root@aswin-VirtualBox: /home/aswin#
```

- Next, you'll need to make sure you have a compiler available. Run this command to install build-essential:

A terminal window titled 'root@aswin-VirtualBox: /home/aswin' showing the output of the command 'sudo apt-get install build-essential -y'. The output indicates that build-essential is already the newest version and lists several packages that were automatically installed and are no longer required.

```
root@aswin-VirtualBox: /home/aswin# sudo apt-get install build-essential -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.8ubuntu1.1).
The following packages were automatically installed and are no longer required:
  linux-headers-5.8.0-55-generic linux-hwe-5.8-headers-5.8.0-55 linux-image-5.8.0-55-generic linux-modules-5.8.0-55-generic
  linux-modules-extra-5.8.0-55-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 76 not upgraded.
root@aswin-VirtualBox: /home/aswin#
```

Step 2: Download Dependencies

- When installing a package from source code, you'll need to manage the installation of the package dependencies. We'll use apt-get to install git's dependencies:

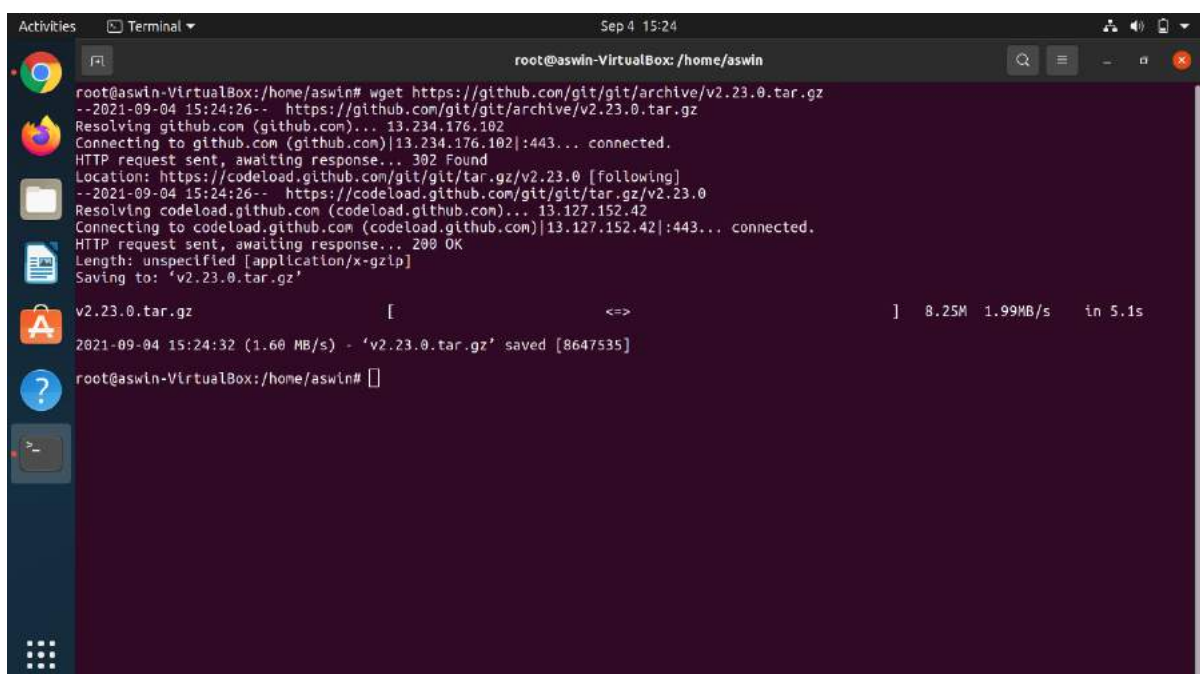


A terminal window titled 'Terminal' with a date and time of 'Sep 4 15:21'. The prompt is 'root@aswin-VirtualBox: /home/aswin'. The command entered is 'apt install build-essential dh-autoreconf libcurl4-gnutls-dev libexpat1-dev gettext libz-dev libssl-dev -y'. The command is partially executed, with a cursor at the end of the line.

```
root@aswin-VirtualBox: /home/aswin# apt install build-essential dh-autoreconf libcurl4-gnutls-dev libexpat1-dev gettext libz-dev libssl-dev -y
```

Step 3: Download the Source Package

- Once the package dependencies are in place, it's time to download the package with wget:



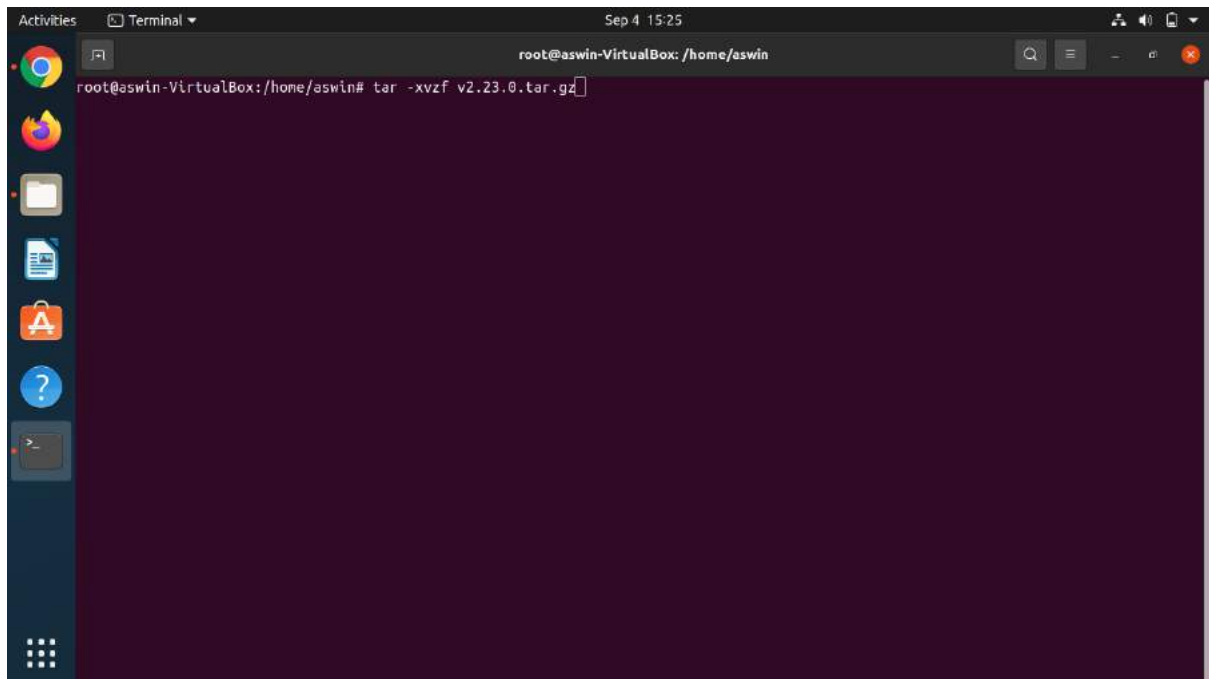
A terminal window titled 'Terminal' with a date and time of 'Sep 4 15:24'. The prompt is 'root@aswin-VirtualBox: /home/aswin'. The command entered is 'wget https://github.com/git/git/archive/v2.23.0.tar.gz'. The output shows the download progress, including the location of the file, the connection status, and the final saved file name 'v2.23.0.tar.gz'.

```
root@aswin-VirtualBox: /home/aswin# wget https://github.com/git/git/archive/v2.23.0.tar.gz
--2021-09-04 15:24:26-- https://github.com/git/git/archive/v2.23.0.tar.gz
Resolving github.com (github.com)... 13.234.176.102
Connecting to github.com (github.com)|13.234.176.102|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/git/git/tar.gz/v2.23.0 [following]
--2021-09-04 15:24:26-- https://codeload.github.com/git/git/tar.gz/v2.23.0
Resolving codeload.github.com (codeload.github.com)... 13.127.152.42
Connecting to codeload.github.com (codeload.github.com)|13.127.152.42|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: 'v2.23.0.tar.gz'

v2.23.0.tar.gz           [          ] 8.25M  1.99MB/s  in 5.1s
2021-09-04 15:24:32 (1.60 MB/s) - 'v2.23.0.tar.gz' saved [8647535]

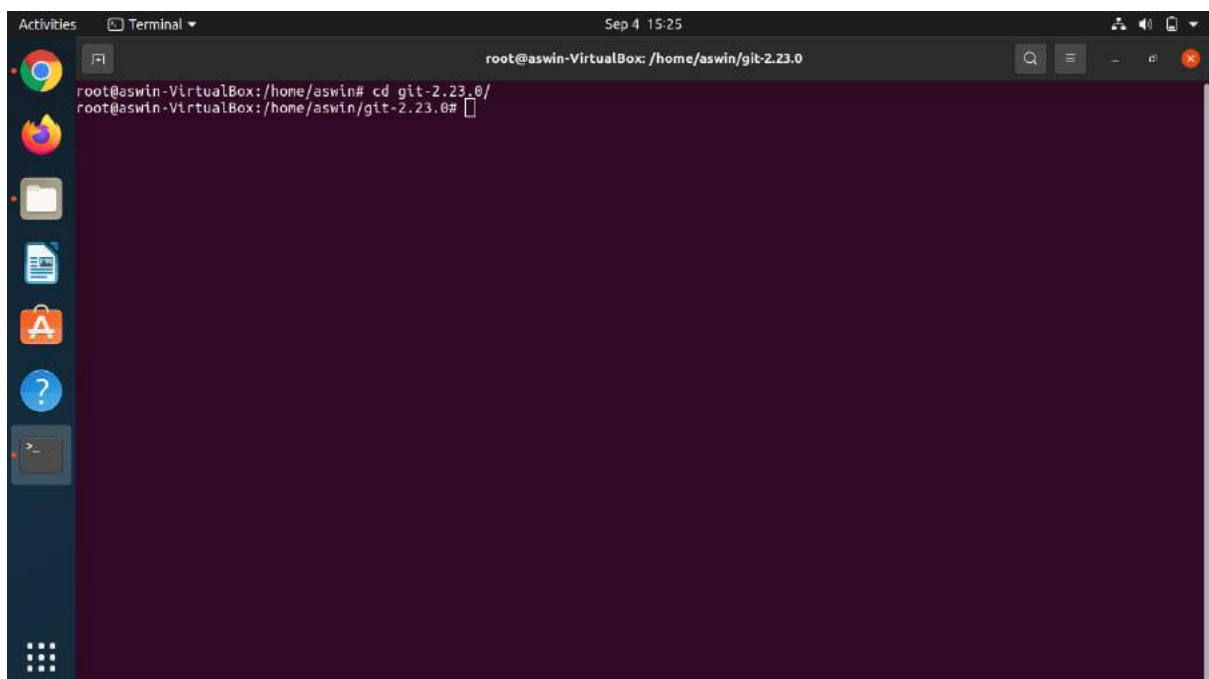
root@aswin-VirtualBox: /home/aswin#
```

- Next, we need to extract the archive and cd (change directories) into the new git directory:



A terminal window titled "Terminal" with a date and time of "Sep 4 15:25". The window shows the command `tar -xvzf v2.23.0.tar.gz` being executed in the directory `/home/aswin`. The prompt is `root@aswin-VirtualBox: /home/aswin#`. The terminal has a dark purple background and a sidebar on the left with various application icons.

```
root@aswin-VirtualBox: /home/aswin# tar -xvzf v2.23.0.tar.gz
```

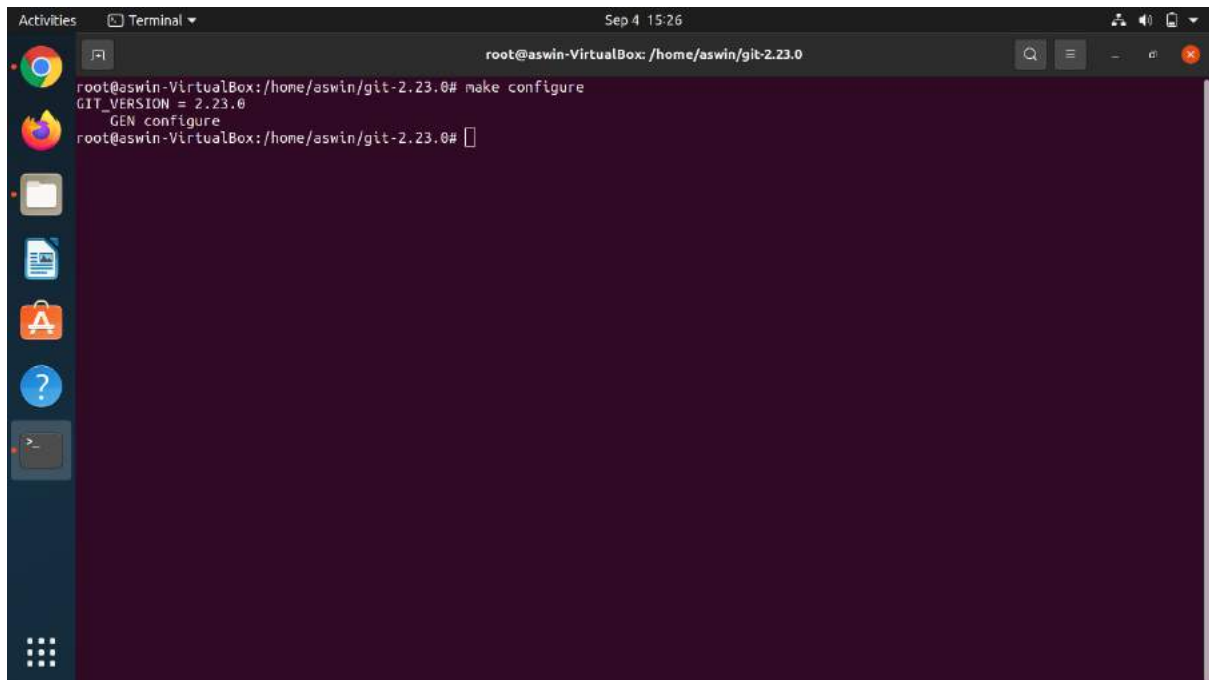


A terminal window titled "Terminal" with a date and time of "Sep 4 15:25". The window shows the command `cd git-2.23.0/` being executed in the directory `/home/aswin`. The prompt is `root@aswin-VirtualBox: /home/aswin#`. The terminal has a dark purple background and a sidebar on the left with various application icons.

```
root@aswin-VirtualBox: /home/aswin# cd git-2.23.0/
root@aswin-VirtualBox: /home/aswin/git-2.23.0#
```

Step 4: Install Git

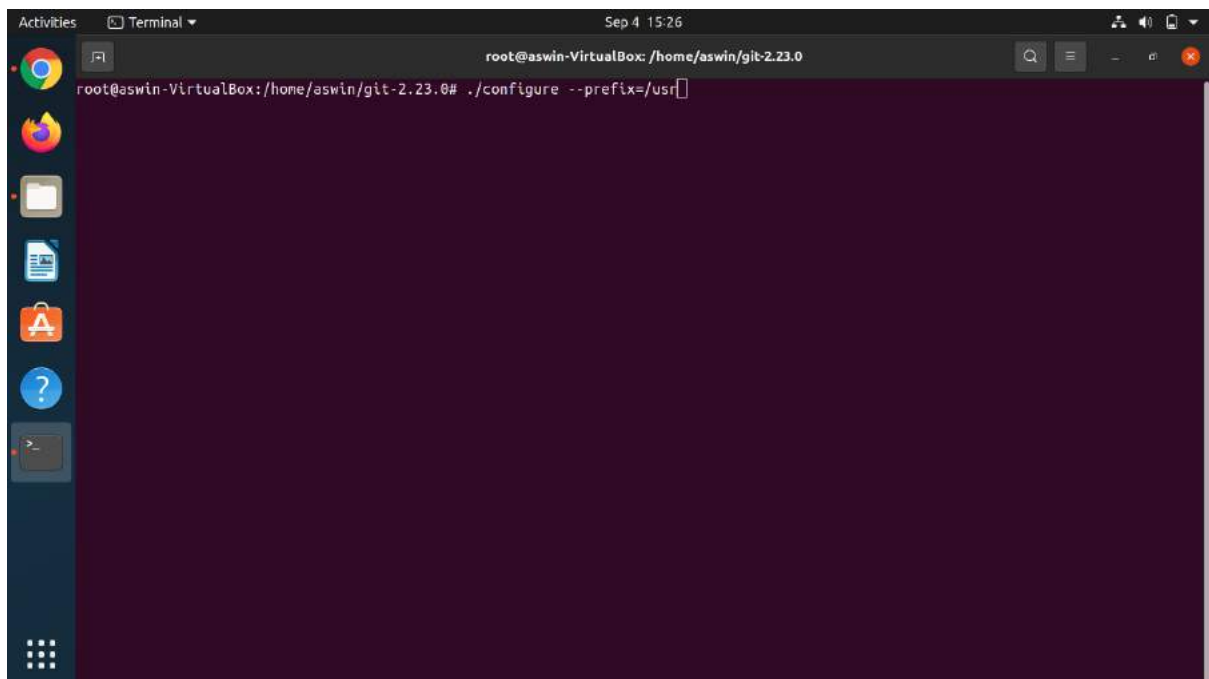
- Now that we have our package extracted and ready to go, we need to configure it:



A terminal window titled "Terminal" with a dark background. The prompt is "root@aswin-VirtualBox: /home/aswin/git-2.23.0". The user has entered the command "make configure". The output shows "GIT_VERSION = 2.23.0" and "GEN configure". The prompt is now "root@aswin-VirtualBox: /home/aswin/git-2.23.0#".

```
root@aswin-VirtualBox: /home/aswin/git-2.23.0# make configure
GIT_VERSION = 2.23.0
GEN configure
root@aswin-VirtualBox: /home/aswin/git-2.23.0#
```

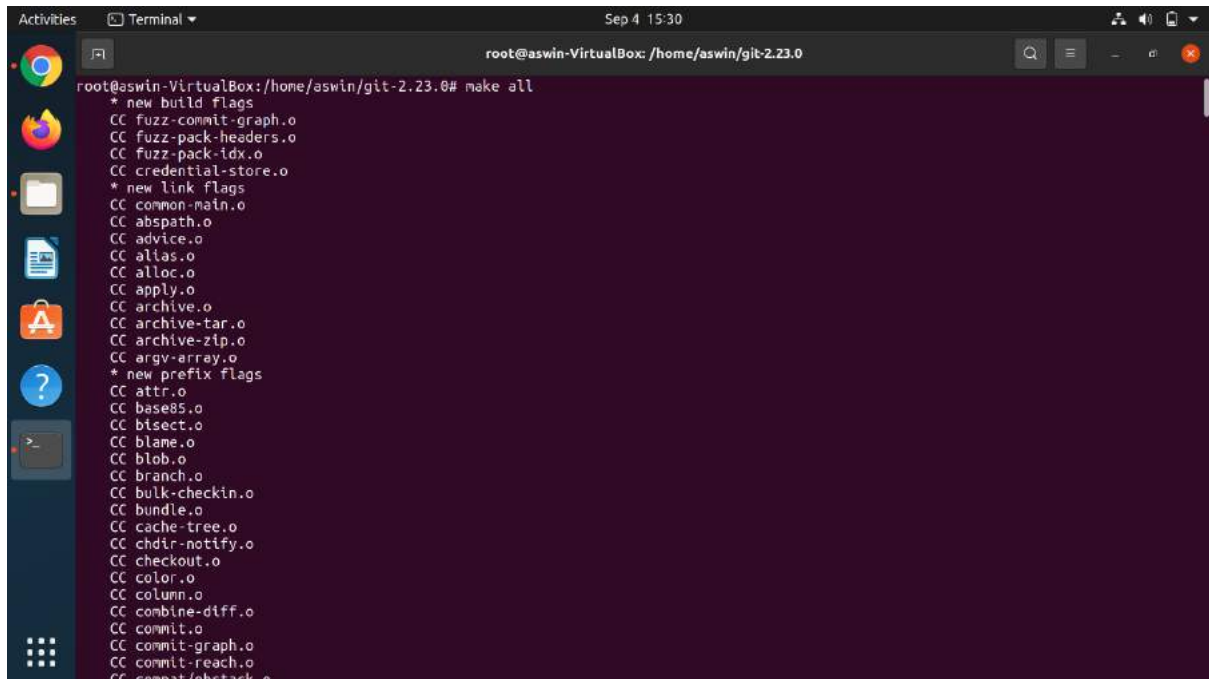
- Next, let's verify that all of the dependencies necessary to build the package are available by running this command:



A terminal window titled "Terminal" with a dark background. The prompt is "root@aswin-VirtualBox: /home/aswin/git-2.23.0". The user has entered the command "./configure --prefix=/usr". The prompt is now "root@aswin-VirtualBox: /home/aswin/git-2.23.0# ./configure --prefix=/usr".

```
root@aswin-VirtualBox: /home/aswin/git-2.23.0# ./configure --prefix=/usr
```

- After that, we'll build the source code:

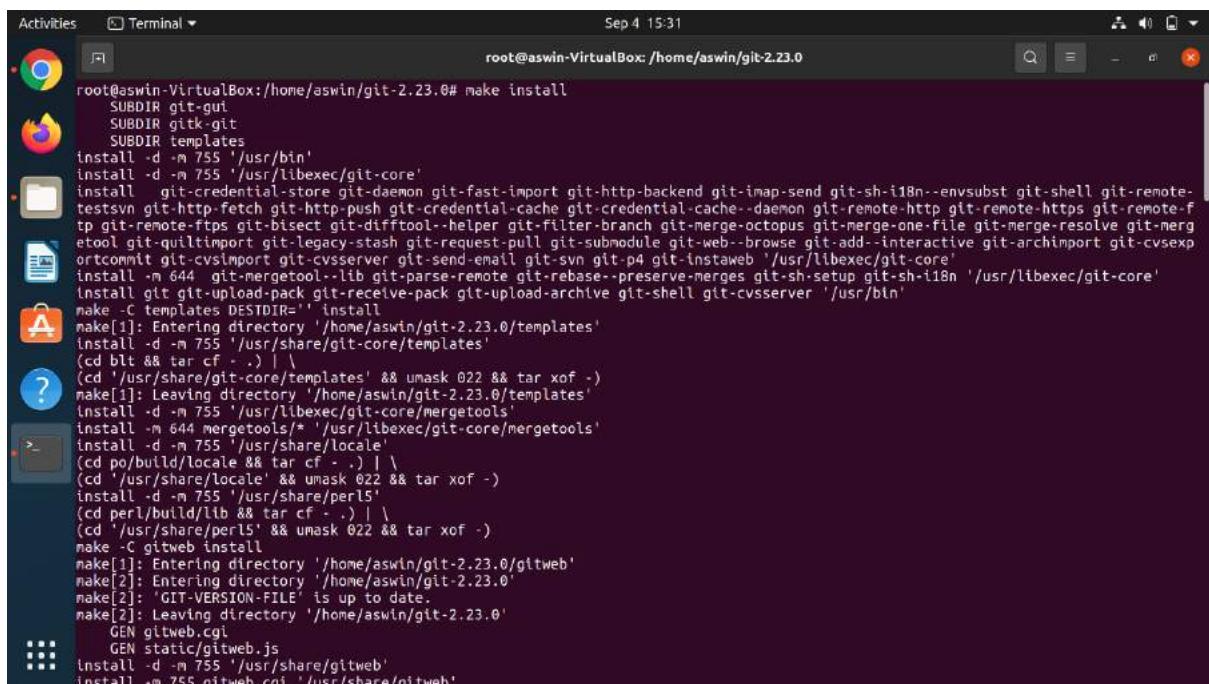


```

root@aswin-VirtualBox:/home/aswin/git-2.23.0# make all
* new build flags
CC fuzz-commit-graph.o
CC fuzz-pack-headers.o
CC fuzz-pack-idx.o
CC credential-store.o
* new link flags
CC common-main.o
CC abspath.o
CC advice.o
CC alias.o
CC alloc.o
CC apply.o
CC archive.o
CC archive-tar.o
CC archive-zip.o
CC argv-array.o
* new prefix flags
CC attr.o
CC base85.o
CC bisect.o
CC blame.o
CC blob.o
CC branch.o
CC bulk-checkin.o
CC bundle.o
CC cache-tree.o
CC chdir-notify.o
CC checkout.o
CC color.o
CC column.o
CC combine-diff.o
CC commit.o
CC commit-graph.o
CC commit-reach.o
CC compat/objstack.o

```

- Now that the binaries are all built, its time to install git:

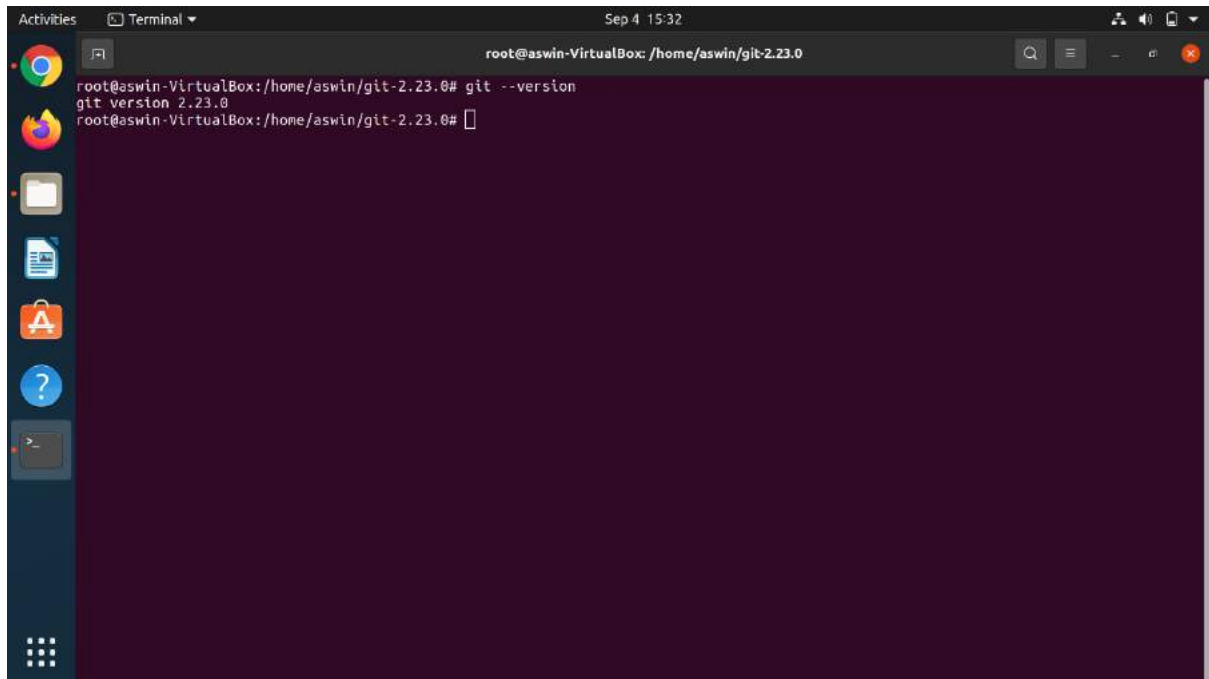


```

root@aswin-VirtualBox:/home/aswin/git-2.23.0# make install
SUBDIR git-gui
SUBDIR gitk-git
SUBDIR templates
install -d -m 755 '/usr/bin'
install -d -m 755 '/usr/libexec/git-core'
install -git-credential-store git-daemon git-fast-import git-http-backend git-imap-send git-sh-i18n--envsubst git-shell git-remote-
tests-vn git-http-fetch git-http-push git-credential-cache git-credential-cache--daemon git-remote-http git-remote-https git-remote-f
tp git-remote-ftp git-bisect git-difftool--helper git-filter-branch git-merge-octopus git-merge-one-file git-merge-resolve git-merg
etool git-quiltimport git-legacy-stash git-request-pull git-submodule git-web--browse git-add--interactive git-archimport git-cvsexp
ortcommit git-cvsiimport git-cvsserver git-send-email git-svn git-p4 git-instaweb '/usr/libexec/git-core'
install -m 644 git-mergetool--lib git-parse-remote git-rebase--preserve-merges git-sh-setup git-sh-i18n '/usr/libexec/git-core'
install git git-upload-pack git-receive-pack git-upload-archive git-shell git-cvsserver '/usr/bin'
make -C templates DESTDIR=' install
make[1]: Entering directory '/home/aswin/git-2.23.0/templates'
install -d -m 755 '/usr/share/git-core/templates'
(cd blt && tar cf - .) | \
(cd '/usr/share/git-core/templates' && umask 022 && tar xof -)
make[1]: Leaving directory '/home/aswin/git-2.23.0/templates'
install -d -m 755 '/usr/libexec/git-core/mergetools'
install -m 644 mergetools/* '/usr/libexec/git-core/mergetools'
install -d -m 755 '/usr/share/locale'
(cd po/build/locale && tar cf - .) | \
(cd '/usr/share/locale' && umask 022 && tar xof -)
install -d -m 755 '/usr/share/perl5'
(cd perl/build/lib && tar cf - .) | \
(cd '/usr/share/perl5' && umask 022 && tar xof -)
make -C gitweb install
make[1]: Entering directory '/home/aswin/git-2.23.0/gitweb'
make[2]: Entering directory '/home/aswin/git-2.23.0'
make[2]: 'GIT-VERSION-FILE' is up to date.
make[2]: Leaving directory '/home/aswin/git-2.23.0'
GEN gitweb.cgi
GEN static/gitweb.js
install -d -m 755 '/usr/share/gitweb'
install -m 755 gitweb.cgi '/usr/share/gitweb'

```

- That's it! The last thing to do is to verify that git is working:

A screenshot of a Linux terminal window. The window title is "Terminal" and the current directory is "/home/aswin/git-2.23.0". The prompt is "root@aswin-VirtualBox:". The user has entered the command "git --version" and the output is "git version 2.23.0". The terminal has a dark purple background and a light blue cursor. The window is part of a desktop environment with a sidebar on the left containing icons for various applications like a web browser, file manager, and terminal. The top bar shows the date and time as "Sep 4 15:32".

```
root@aswin-VirtualBox: /home/aswin/git-2.23.0
root@aswin-VirtualBox:/home/aswin/git-2.23.0# git --version
git version 2.23.0
root@aswin-VirtualBox:/home/aswin/git-2.23.0#
```