

# Configure Ubuntu for Software Development

Ronald Mak

Department of Computer Engineering

Department of Computer Science

Department of Applied Data Science

January 11, 2020

Revised: February 16, 2020

## Introduction

This tutorial assumes you've installed Ubuntu, either directly on Windows 10 as Windows Subsystem for Linux, or on a virtual machine hosted by VirtualBox running on MacOS X or Windows 10. See the tutorial "Install Ubuntu on Windows 10 and on Virtual Box":

<http://www.cs.sjsu.edu/~mak/tutorials/InstallUbuntu.pdf>

## Install developer tools

An Ubuntu distribution does not include developer tools such as the GNU **gcc** and **g++** compilers and GNU **make**, so you must manually install them.

Open a terminal window in Ubuntu and enter the following commands:

```
sudo apt update
sudo apt upgrade
sudo apt install build-essential
```

(Use **apt-get** for older versions of Ubuntu before 16.04.) See

<https://help.ubuntu.com/community/InstallingCompilers>

**Note:** If you receive the error message "Temporary failure resolving us.archive.ubuntu.com", the fix is to enter the following command:

```
sudo dhclient -v -4
```

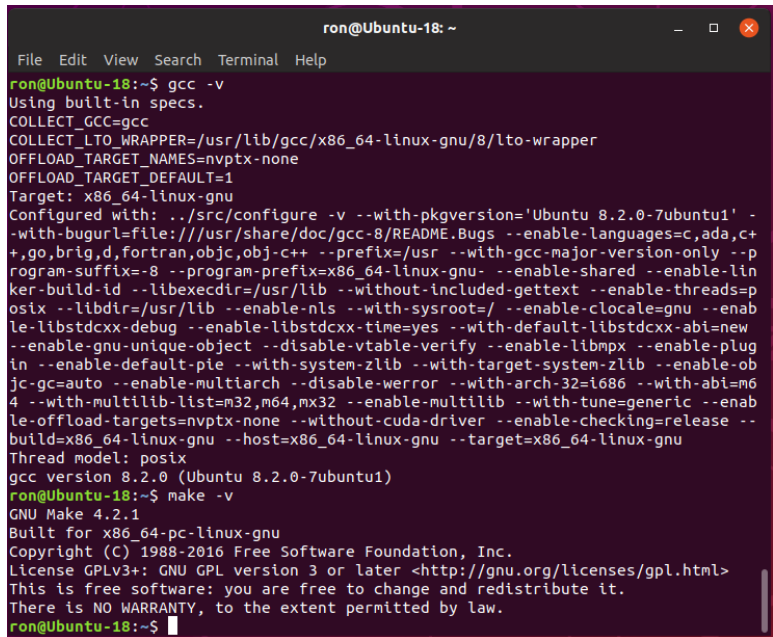
and then try again. See <https://askubuntu.com/questions/884604/temporary-failure-resolving-us-archive-ubuntu-com-live-usb-rescue>.

## R. Mak, Configure Ubuntu for Software Development

After installing the developer tools, you should be able to verify that gcc, g++, and make were properly installed by entering the following commands in the Ubuntu terminal (Figure 1):

```
gcc -v
g++ -v
make -v
```

Figure 1. Verify that gcc, g++, and make were properly installed.



```
ron@Ubuntu-18: ~
File Edit View Search Terminal Help
ron@Ubuntu-18:~$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-linux-gnu/8/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-linux-gnu
Configured with: ../src/configure -v --with-pkgversion='Ubuntu 8.2.0-7ubuntu1' --with-bugurl=file:///usr/share/doc/gcc-8/README.Bugs --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++ --prefix=/usr --with-gcc-major-version-only --program-suffix=-8 --program-prefix=x86_64-linux-gnu- --enable-shared --enable-linker-build-id --libexecdir=/usr/lib --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --with-sysroot=/ --enable-clocale-gnu --enable-libstdcxx-debug --enable-libstdcxx-time=yes --with-default-libstdcxx-abi=new --enable-gnu-unique-object --disable-vtable-verify --enable-libmpx --enable-plugin --enable-default-pie --with-system-zlib --with-target-system-zlib --enable-objc-gc=auto --enable-multiarch --disable-werror --with-arch=32=i686 --with-abi=m64 --with-multilib-list=m32,m64,mx32 --enable-multilib --with-tune=generic --enable-offload-targets=nvptx-none --without-cuda-driver --enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu
Thread model: posix
gcc version 8.2.0 (Ubuntu 8.2.0-7ubuntu1)
ron@Ubuntu-18:~$ g++ -v
GNU Make 4.2.1
Built for x86_64-pc-linux-gnu
Copyright (C) 1988-2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
ron@Ubuntu-18:~$
```

## Useful Ubuntu utility programs

The GUI-based file browser named **Nautilus** allows you to view and manipulate files in the Ubuntu file system. If it is not already installed, enter the following command into the Ubuntu terminal window:

```
sudo apt install nautilus
```

To run Nautilus, open another Ubuntu terminal window and enter the command

```
nautilus
```

(You want to run this in another terminal window because you want Nautilus to run as a separate process. Also, Nautilus outputs warning messages that you can usually ignore but are distracting if they appear in your main terminal window.)

The text editor named **Gedit**, provides a GUI to create and edit text files. If it is not already installed, enter the following command into the Ubuntu terminal window:

```
sudo apt install gedit
```

To run Gedit and edit an existing text file, enter the `gedit` command followed by the name of the file, or just `gedit` if you want to create a new text file. For example:

```
gedit .bashrc
```