Linux Beginner Guide

Jaewoong Lee

Ulsan National Institute of Science and Technology

jwlee230@unist.ac.kr

January 5, 2020

Introduction

In this guide, I assume that followings are already installed:

1. Ubuntu 16.04.2 or Higher
2. ZSH 5.0.2 or Higher
3. VIM 8.1 or Higher
4. We will connect to server via SSH

With this guide, you can use and understand Linux system.
Also, this guide includes as little information about operating system as possible. If you find some fault in the strict sense of the word, that means you are not **beginner**.

Overview

1. Linux?
2. Basic Linux Command
3. Edit File with VIM
4. IO Redirections

Linux?

Figure: Linus Torvalds, Inventor of Linux

Linux is one of the most famous OS as Windows and macOS.
Linux is open-source project.
Android, OS for mobile, is based on Linux.

Ubuntu?

Figure: Logo of Ubuntu

Ubuntu is an OS which is based on Linux.
Ubuntu is the best OS in Linux-like OS, because of convenience of its installation and usage.

Where we start

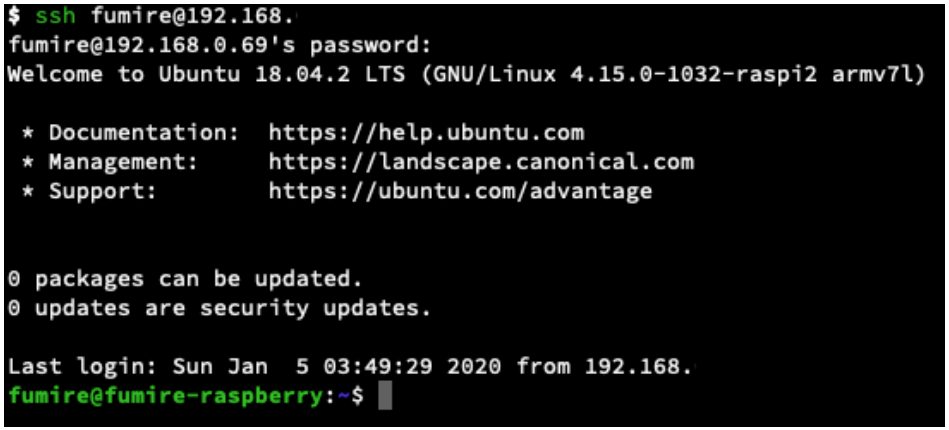

Figure: Here is where we start

After you connect to server via SSH, you can see like this.
Here is where we start!
fumire will be user name, and *fumire-raspberry* will be server name.

pwd

```
fumire@fumire-raspberry:~$ pwd
/home/fumire
```

Figure: Result of *pwd* Command

pwd is abbr. of "Print Working Directory".
You can see where you are with *pwd* command.
Also, "/home/username" is your *home folder*, a.k.a. '∼'.

ls

```
fumire@fumire-raspberry:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
```

Figure: Result of *ls* Command

ls stands for "List".
ls command lists current directory contents.
If current directory is empty, the result will be nothing.

Configuration

However, you have not completed configuration. Therefore, finish settings with following command:

Example

```
$ git clone https://github.com/Fumire/.dotfiles.git
$ cd .dotfiles
$ make
$ chsh -s /usr/bin/zsh
```

Note that you should input command only after '$'.
After executing commands, you should restart your shell.

Configuration *(Cont.)*

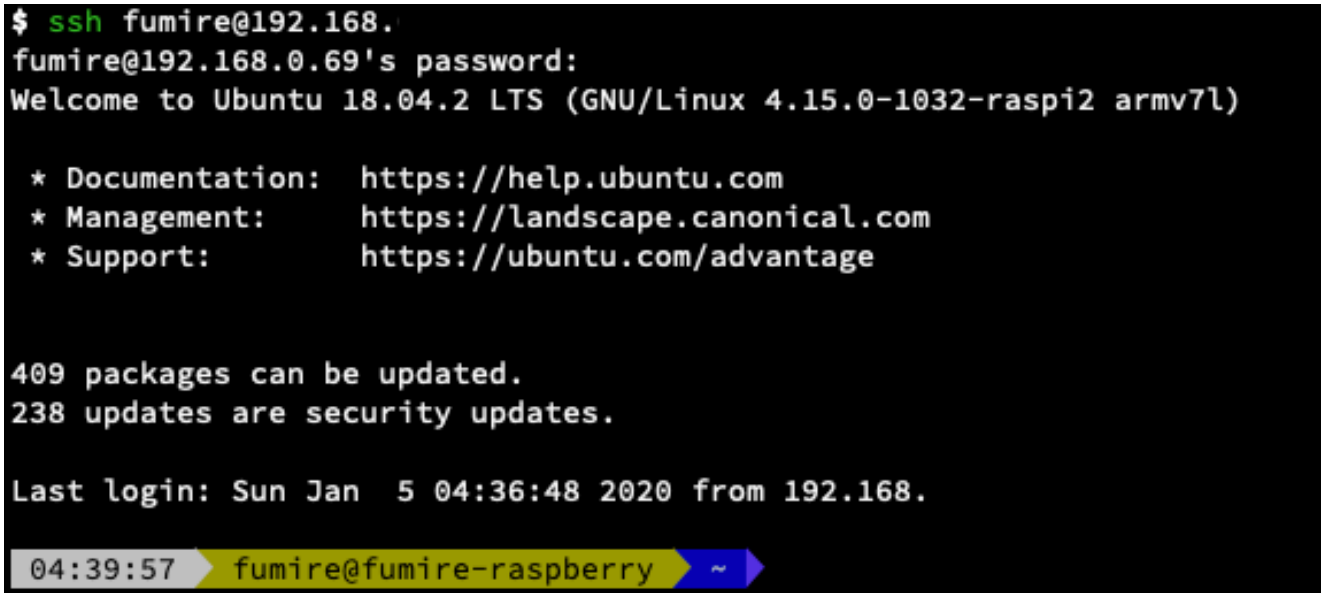

Figure: ZSH

With successful configuration, you can see like this.

Tip!

Figure: Right Command vs. Wrong Command

You can easily know this command is right with ZSH as figure.

mkdir

mkdir stands for "Make Directory".
You can make a directory which named 'test' as following:

Example

$ mkdir test
or
$ md test

mkdir returns nothing. Literally, *mkdir* command only make directory.
You can check that the directory has been made with *ls* command.

cd

cd is abbr. of "Change Directory".
You can change your working directory to 'test' as following:

Example

```
$ pwd
$ cd test
$ pwd
```

Also, you can go your home folder at once with *cd*, no matter where you are.

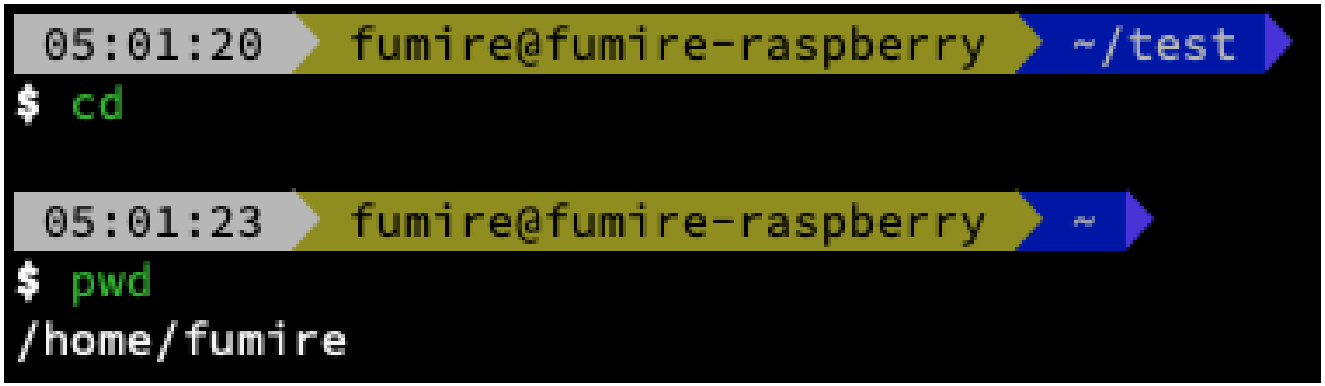

Figure: *cd* will guide you to home folder

Tip!

If you hit "Tab" button, ZSH will give proper candidates.
Following example shows what ZSH gives.

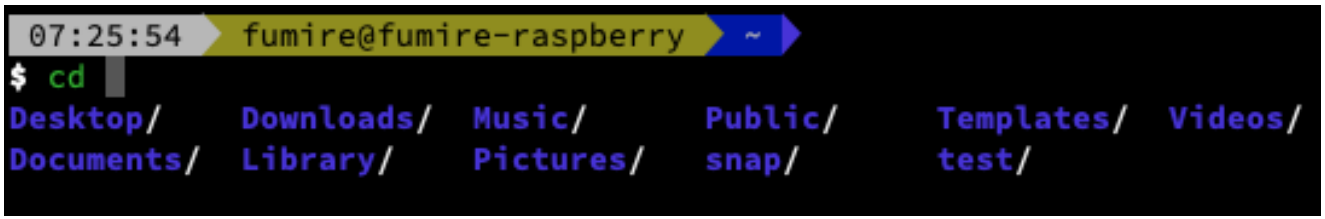

Figure: Shortcut with Tab

man and --help

You can get detailed information about command as following:

Example

$ man ls
and/or
$ ls --help

This guide will give simple information about Linux command. Hence, when you have curiosity about command, use these command.

Directory Structure

Try following commands:

Example

```
$ cd test
$ ls -al
```

Then, you can see like this:

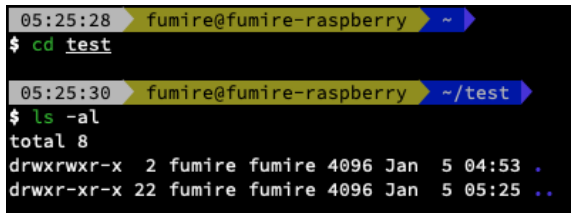

Figure: Result of *ls* command

All directory has '.' and '..', even though the directory is empty.
'.' means current directory itself; and, '..' means parent directory.

touch

touch command make new file or touch the file.
Try following example:

Example

```
$ cd
$ touch t
$ ls
```

Then, you can see that the file which name 't' has been made.

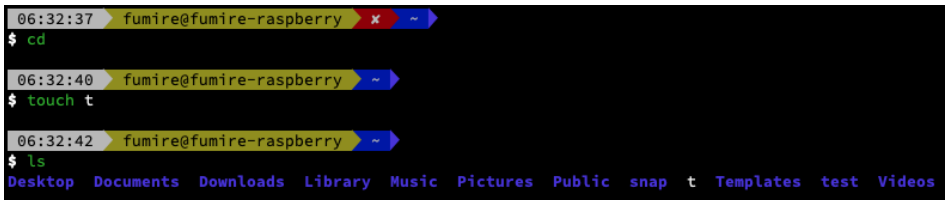

Figure: Result of *touch* Command

mv

mv command moves/renames file. *mv* is used as:

Example

$ mv SRC(source) DST(destination)

Try following commands:

Example

$ mv t tmp
$ ls
$ mv tmp test/
$ ls

Then, you will realize that the file 'tmp' is gone. I hope that you already know where the file goes. :)

cp

cp command copies SRC to DST. *cp* is used as:

Example

```
$ cp SRC DST
```

Try following commands:

Example

```
$ cd ~/test/
$ ls
$ cp tmp tmp2
$ ls
```

Then, you can realize that a new file 'tmp2' has been made.

sudo

sudo is abbr. of "Substitute User do"; but, many people know as "Super User do".
sudo allows a system administrator to delegate authority to give certain user the ability to run some command as another user.

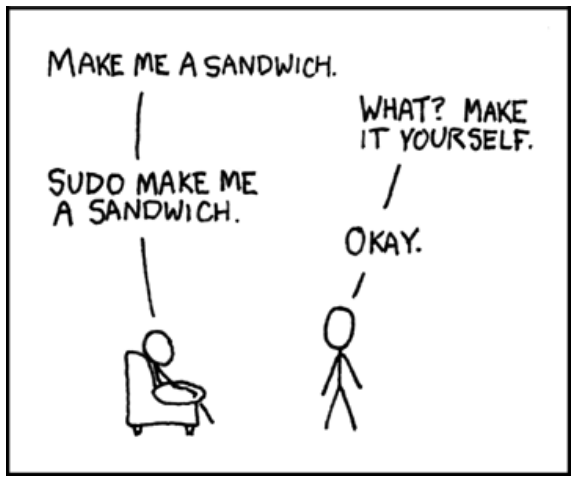

Figure: XKCD: Sandwich

THINK what will happen after *sudo* command!!

Editor

There are three major editors in Linux.

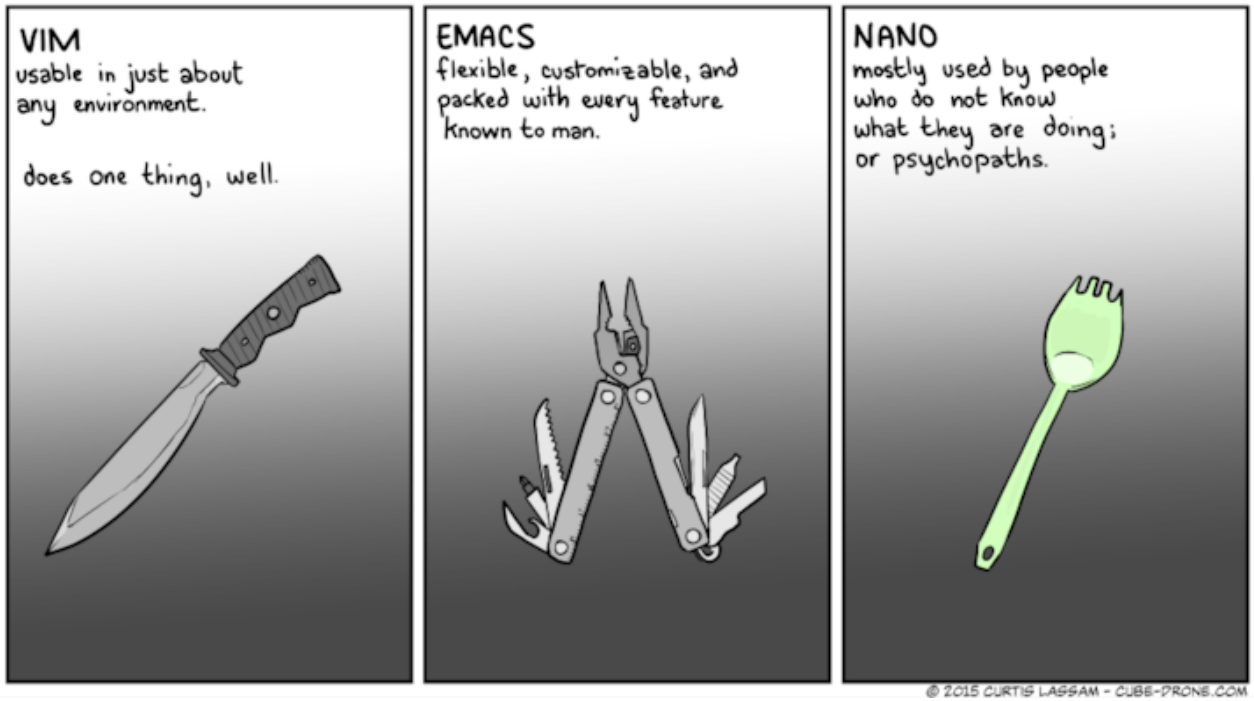

Figure: Descriptions of Editor

For this reason, this guide use VIM editor.

Editor *Cont.*

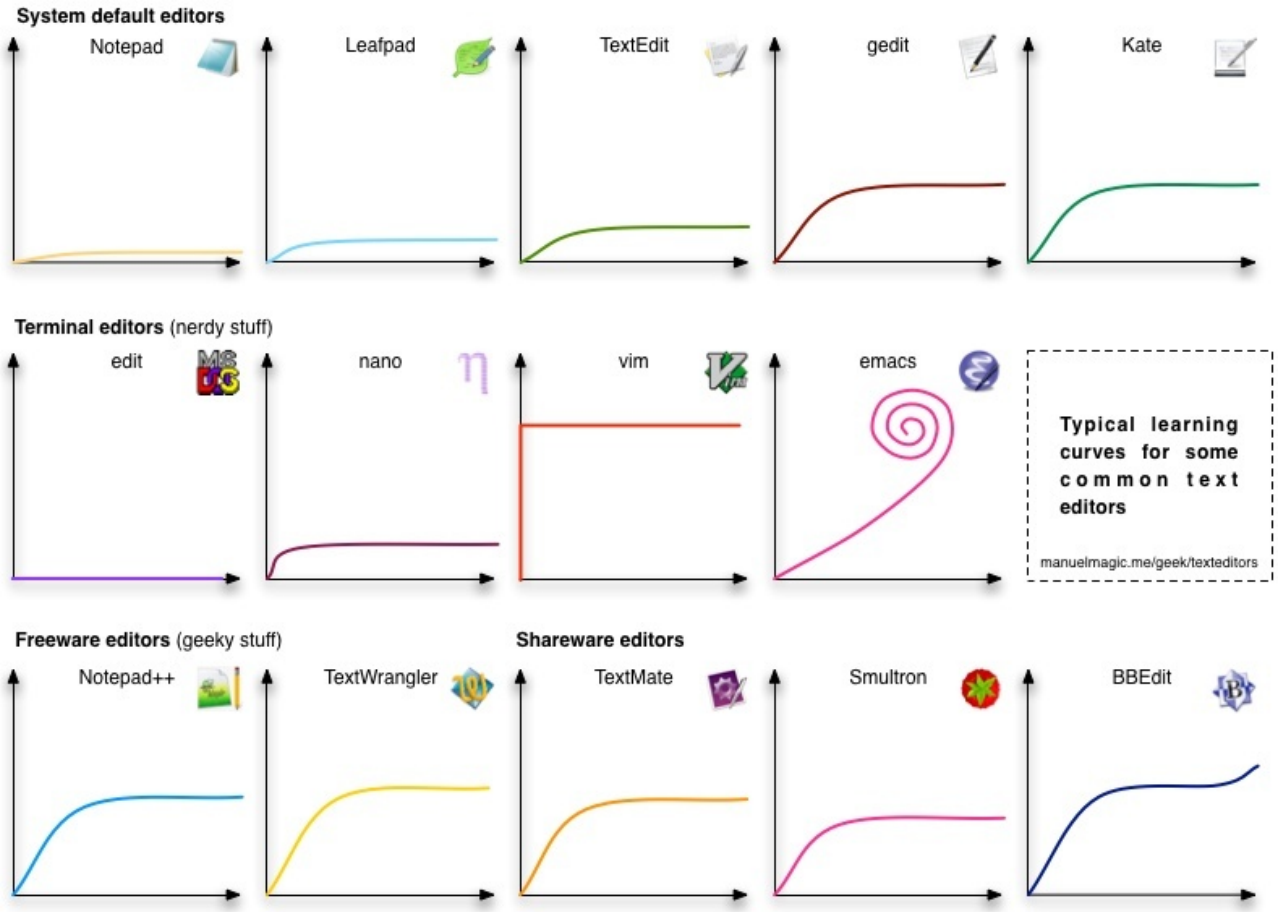

Figure: Learning Curves among Editors

First Meet with VIM

With these commands, you can make/edit file.

Example

```
$ vi tmp
```

If it is first time to open VIM, then you will see like this.

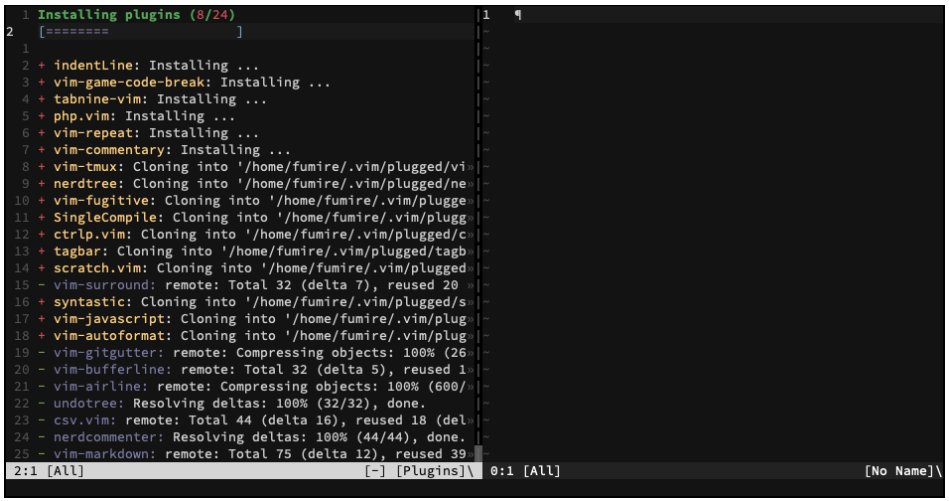

Figure: First Time of VIM

Modes of VIM

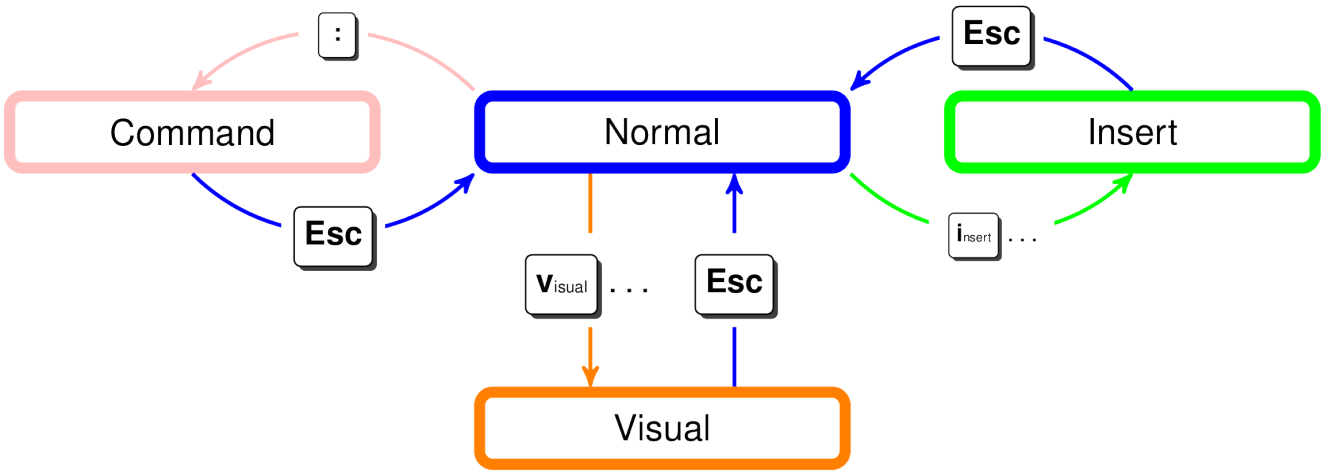

Figure: Three Modes in VIM

How to Edit with VIM

Editing with VIM is following such steps:

1. Press 'i'
2. Edit the file
3. Press 'ESC'
4. Enter ':w' which means *write*
5. Enter ':q' wihch means *quit*

Plugin Setting (Optional)

You might see the error message because the plugin setting is not completed. To solve this, use following commands:

Example

```
$ cd ~/.vim/plugged/tabnine-vim/
$ python3 install.py
```

Then, all plugin acts properly without errors.

cat

cat stands for con**cat**enate. *cat* command reads files, and writing them to standard output.
Consider following example:

Example

```
$ cd ~/test/
$ cat tmp
```

You can see contents of file.

Output to file

If you want redirect output to file, use following example:

Example

```
$ cat tmp > output
```

However, this method *overwrites* the contents of file.
If you want preserve the file contents, use following:

Example

```
$ cat tmp >> output
```

Output to file *Cont.*

In some cases, you should divide standard output and standard error.
In these cases, use following commands:

Example

```
$ commands 1> STDOUT 2> STDERR
```

Example

```
$ cat tmp 1> ex.stdout 2> ex.stderr
```

more / less

more and *less* are commands for seeing the contents of file. Consider following examples:

Example

```
$ more tmp
```

Example

```
$ less tmp
```

Pipe

Use pipe (|) to indicate input as output of previous command.

Example

```
$ command1 | command2
```

The output of command 1 will be the input of command2.
Consider following example:

Example

```
$ cat tmp | less
```