**Continue Setup**

Justice Escalante

Linux Admin

**Checking your repositories**

Unlike **CentOS** or some of the other distros, viewing the repositories can be completely different than each other as well as its location. To view repositories **in Ubuntu** you can use the command ***cat/etc/apt/sources.list*** to find out all of what’s included, for the enabled only repositories, you can use ***apt-cache policy***. ***/etc/apt/sources.list*** is the repository location for this distro as of 4/26/23. For **Ubuntu** adding repositories is rather quite simple compared to other distros. Editing repositories as well can be done with a text editor such as **vim**, once done editing update the modified repository. It’s also helpful to make a copy of this repository just in case something goes wrong. In Ubuntu a simple command such as ***sudo add-apt-repository insertname*** would do the trick for official repositories, simply swap out ***insertname*** with what repository you want. Make sure before adding anything we update the list before hand, this can also be done by ***sudo apt update***. Now for PPA repositories it’s a little extra onto what would be needed for officials as we will use the package ***software-properties-common*** to help. Once this is installed with ***sudo apt-get install software-properties-common***: we can now use the command ***sudo add-apt-repository PPA:<PPANAME>/<PPA>*** to add a PPA repository of our liking.

Text

Description automatically generated

**Installing Tmux, Fail2ban, and emacs**

*As this distro is more consumer friendly, setting up programs is a lot easier as for most you can install with* ***sudo apt install PACKAGENAME****. However, you could also install other package managers like* ***snap rubygems*** *etc if it fits your preference.*

**fail2ban**

Installing with ***sudo apt install fail2ban***

Text

Description automatically generated

Above you can see the screenshot regarding **fail2ban**being installed, its key to pay attention/read when doing installs because of extra stuff that might be needed/potentially snucked in. After installation on 4/27/23. **Fail2ban** has plenty of options to control its log in, these include how many retries(***maxretrys***) and other cool stuff like ban ranges(***bantime***).

To edit **fail2ban** first find the file called ***jail.con***f located at ***/etc/fail2ban/***, for extra protection a copy of the default conf file is not bad idea if something messes up. Now you can mess around with parameters like **maxretry**, For more commands: <https://www.fail2ban.org/wiki/index.php/Commands>

Text

Description automatically generated

After swapping some settings out, we can now enable and start **fail2ban** with ***sudo systemctl enable fail2ban*** and ***sudo systemctl start fail2ban***.

A picture containing text

Description automatically generated

Above you can see we have tested to see if f**ail2ban** will run and its status, which is important as not all programs run on install some manual. Now to fully test if **fail2ban** works you could stress test on a vpn or another pc. To see who’s been banned, the file with this information would be located at ***/var/lod/fail2ban.og***. Now to find the bans we would have to use **grep** with **‘Ban’** as that file has more then just that. If you want to see all your packages that have been installed also to check use ***apt list –installed***.

**Emacs**

Installing with ***sudo apt install emacs***

Text

Description automatically generated

To test if **emacs** work properly, we can use its format to open a file: ***emacs FILENAME***. Below the command ***Ctrl+s*** was used to be able to search in the file, as you can see below, we hid “test” around the file and emacs found it easily.

Graphical user interface, text, application

Description automatically generated

Now to quit **emacs**, hold ***Ctrl*** followed by a ***x*** and then a ***c***, it will prompt you to save with a little screen below. This is important!



**Tmux**

Installing **Tmux** ***with sudo apt install tmux***

Text

Description automatically generated

Now to check if **tmux** is working properly, we can start a new session with the format of **tmux new -s INSERTSESSIONNAME**. To add tabs in **tmux** simply do ***Ctrl+b*** followed shortly with a **c**, once these tabs are open you can navigate with ***Ctrl+b*** as well but followed by the number of the tab. You can also use ***tmux new-window*** to create a new window as well inside **tmux**. Below is what **tmux** should look like if done properly.

Graphical user interface, text, application, chat or text message

Description automatically generated

To quit, type in ***:kill-session***, ***tmux detach*** in **tmux** or ***Ctrl+D*** should work as detachment. Later on if you want to rejoin that session, simple ***use tmux attach -t INSERTSESSIONAME***

**How to install fun tools such as lolcat and cowsay**

*Installing these programs with* ***sudo apt install*** *was surprisingly quite easy as these had minimal setup. Once installed, both were tested to make sure they work properly as seen below. This was done and is working on 4/27/23.*

**lolcat**

Installing **lolcat** with ***sudo apt install lolcat***

Text

Description automatically generated

Now a great way to see if **lolcat** functions properly is to run it! For this example, below we kept putting the command ***echo “rainbow” | lolcat*** to see all the different colors it could use.

Graphical user interface

Description automatically generated

**Cowsay**

Installed with ***sudo apt install cowsay***

Text

Description automatically generated

Testing **cowsays** function with ***cowsay “Continue my setup”***

Text

Description automatically generated

**cowsay** also has a lot of other animals that can be used such as turkeys, a hotdog character etc. More animals here: <https://linuxhint.com/cowsay-linux-command/>

A picture containing graphical user interface

Description automatically generated

Installing these programs with ***sudo apt install*** was surprisingly quite easy as these had minimal setup. Once installed, both were tested to make sure they work properly as seen in the screenshots above. This was done and is working on 4/27/23.

**Updating VI to vim, if needed.**

Text

Description automatically generated

Upgrading ***VI*** to the improved version vim can come in handy as vim offers a lot more commands customizability and a better ***UI***. However not all systems will run ***VIM*** or support it. To check if vim is installed you can search for its version number, if no program is there you will get Command ‘vim’ not found. You can update vim ***with sudo apt update*** or simply run the installation again with ***sudo apt install vim***. Above you can see I reran the installation of it, but since mine is up to date as of 4/27/23 nothing was done. Below ***is vim –version***

A screenshot of a computer

Description automatically generated with medium confidence

**Bibliography**

**https://www.youtube.com/watch?v=L5SyOGcDpdU**

**https://www.linkedin.com/pulse/funny-commands-linux-rohan-patil#:~:text=Cowsay%20is%20a%20command%20that,with%20an%20animated%20character%20cow.**

**https://www.youtube.com/watch?v=1kicKTbK768**

**https://www.youtube.com/watch?v=NnCaw57FVMU**

**https://medium.com/geekculture/5-different-ways-to-install-software-on-ubuntu-linux-14ae6b95d1d2**

**https://help.ubuntu.com/community/Repositories/Ubuntu**

**https://www.fail2ban.org/wiki/index.php/Main\_Page**

**https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-ubuntu-20-04**

**https://www.youtube.com/watch?v=Y1busy\_dDRQ**