Finding Adding and Modifying Repositories

Overview

Repositories or Personal Package Archive (PPA) are software packages where server functions and programs are located these packages can each have multiple functions which are enabled or disabled, we can use repositories to add modify or remove functions from the server.

CentOS

To add and modify Repositories in CentOS first make sure you have root privileges you can do this by making yourself Admin before install but if not refer to part 1 on how to do that. You need to locate where they are installed which are **/etc/yum.repos.d** you can **cd** to go to that file path or **ls -a** to see all files in that location

A screen shot of a computer program

Description automatically generated

This only shows what repositories are installed but **enabled** to see which repositories are enabled first Install **dnf** you can also use **yum** the default software package manager for redhat distros instead but **dnf** is a newer version and is used more so it is recommended. You can do the **sudo yum install dnf**

A screen shot of a computer

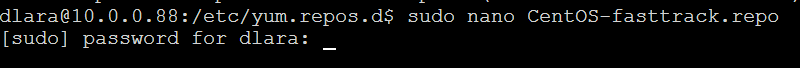
Description automatically generated

We can now use **dnf** for various functions on software packages we can first see what repositories are actually enabled with **sudo dnf** **repolist** to see all the enabled repositories.

A screen shot of a computer

Description automatically generated

If we want to enable a repository we have to modify the contents of each repository file and find where it says enabled change the 0 to one for this instance we will enable the fast track repository which is used to give the server community fixes by opening the **CentOS-fasttrack.repo** file using **sudo nano CentOS-fasttract.repo** make sure to put your user password or root password if you are doing this as root



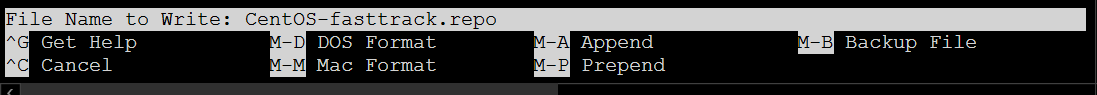
Find where it says enabled change he 0 to a 1 press **ctrl x** to exit then **y** to confirm and then **enter** to finish save process

A computer screen with white text

Description automatically generated

A computer screen shot of a black screen

Description automatically generated



**Press enter when you see this**

To confirm that the repo is modified and enabled we can use the repolist command we did earlier to verify **sudo dnf repolist** mind you it may take a minute or so to display after enabling a repo

A computer screen with white text

Description automatically generated

Make sure to clean any cached information by typing **dnf clean all** in case.

A black background with white text

Description automatically generated

To disable a repo simply go back to the file and change the enabled portion back to 0 then save, exit and use your **sudo dnf** **repolist** to verify it’s disabled.

To install community made repositories it’s always important to follow the instructions provided by the developer there is no strict way of installing repositories so it is important to go by the developer rules but rule of thumbs are usually to

1. Like always make sure to have sudo privileges
2. Install files are formatted in a way of **sudo yum install #repository link** where **sudo** is the privileges needed to do this command **yum** is the manager we are using for software packages you may also use **rpm**, **dnf** other software package managers then any commands and flags like **-y** then the link or mirror of where the repository is being downloaded

In the following example I will install the Harbottle repo I use the command **sudo yum -y install** [**https://harbottle.gitlab.io/harbottle-main/7/x86\_64/harbottle-main-release.rpm**](https://harbottle.gitlab.io/harbottle-main/7/x86_64/harbottle-main-release.rpm) **as shown below**

**A screenshot of a computer program

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

Ubunutu

In order to add or modify a repository in an ubuntu server we first need to figure out where the repositories are located (at least the default ones) which is in the **/etc/apt/sources.list** we can use a **grep** statement to filter out that file to see our enabled repositories **grep ^[^#] /etc/apt/sources.list /etc/apt/sources.list.d** where the **^[^#]** argument filters out comments the file paths are where the information is stored to where we can see our directory.

A computer screen shot of a computer code

Description automatically generated

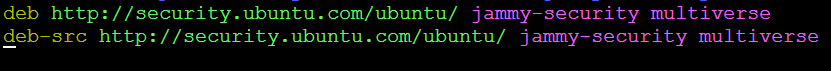
To enable or add a repository we need to modify the **sources.list**  file we can do this by using nano **sudo nano** **etc/apt/sources.list**

A screen shot of a computer screen

Description automatically generated

To enable a repository simply remove a command from the commented lines that start with **deb** then press **ctrl x** then **y** to confirm exit then **y** to finish process. I this example I will uncomment the **deb-src http://us.archive.ubuntu.com/ubuntu/ jammy-updates multiverse** which provides source code for the package above it





After exiting the file update by using the command **sudo apt-get update**

A computer screen shot of a program

Description automatically generated

The process of adding third Party repositories can vary and can be done multiple ways the best way is to follow the instructions but first set the following up if you have not configured IPv6 configurations on your server.

A screen shot of a computer code

Description automatically generated

The following commands disables IPv6 Configurations which is something that the server will use alongside installing any repositories if not installed it can disrupt any installation. While the methods as to how repositories can be installed in ubuntu can vary there are some rule of thumbs first just in case use the **sudo apt-get install software-properties-common** command to make installation of repositories easier then the way installs go is by first getting the repository or security key update then install the repository. In this example I will be installing the **Suricata** PPA and I first get the repository with the command **sudo add-apt-repository ppa:oisf/suricata-stable** as shown below

A screen shot of a computer program

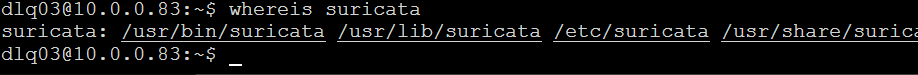
Description automatically generated

Then we need to update to clear any server cache with the command **sudo add-apt update** afterwards we can install with **sudo-apt-get install suricata**

A computer screen shot of a black screen

Description automatically generated

A Make sure to put **y** for yes confirm installation it should then start a long process If there are no error messages you have successfully installed the directory to see where the contents of the repository are installed we can the ls command. We can verify the existence of the directories to see where the files are loaded using the **ls** and **whereis** command



A black background with white text

Description automatically generated

Installing Various Software on both servers

Tmux

Ubuntu

To Install Tmux on Ubuntu use the command **sudo apt-get install tmux** as shown below

A screen shot of a computer

Description automatically generated

Upon putting tmux on the terminal your cli should look something like this

A screen shot of a computer

Description automatically generated

Press **Ctrl + B** then **D** to exit out

A black background with white numbers and symbols

Description automatically generated

Tmux Location

CentOS

To install Tmux first install **epel-release** with the command **yum install epel-release** type **y** when promted

A screenshot of a computer

Description automatically generated

Afterwards the command to install tmux is **sudo yum install tmux** put the password when promted and put **y** to confirm installation if there are no errors you have successfully installed tmux

A screenshot of a computer program

Description automatically generated

Type **tmux** to start a session and **ctrl+b** then d to exit

A screen shot of a computer

Description automatically generated

Location of tmux



Emacs

CentOS

To Install Emacs on CentOS make sure you the **epel-release** repository as the program is located there you can do this with the following command **sudo yum install epel-release**

A screenshot of a computer program

Description automatically generated

If the repository is already installed we can to the next step enter **sudo yum install emacs** or **sudo dnf install emacs** to install the software enter your password and when you see the prompt from the picture below put **y**

Wait a bit for it to install if there are no error messages you have successfully installed emacs

A screen shot of a computer

Description automatically generated

Type **emacs** then the name of the file to make an emacs file



A computer screen shot of a black screen

Description automatically generated

Press **ctrl + x** then **ctrl + c** to then **y** to confirm save and exit

Emacs Location

A black background with white text

Description automatically generated

Ubuntu

The command I used to install Emacs on Ubuntu was **sudo apt-get install emacs** enter the password when promted and type **y** when promted seen on the picture



This will then output a huge swath of text showing the install process if there are no error messages you have successfully installed emacs type **emacs** then the name of the file you want so in this case **emacs testfile**



You should now get something that looks like this

A computer screen shot of a black screen

Description automatically generated

You can add whatever you’d like to save put **ctrl + x** then **ctrl + c** then **y** to confirm save file



Type yes when you see this

A black background with white text

Description automatically generated

Emacs location

Fail2Ban

CentOS

make sure you the **epel-release** for you centOS serverrepository as the program is located there you can do this with the following command **sudo yum install epel-release** (Ignore if already installed)

A screenshot of a computer program

Description automatically generated

Fail2Ban

CentOS

We can install fail2ban on CentOS with the command **sudo dnf install fail2ban** or **sudo yum install fail2ban** enter the password and type **y** when promted if there are no error messages you have successfully installed fail2ban but you need to configure it by creating a local jail.

A computer screen with a black background

Description automatically generated

**Enabled, port and filter** are default network settings for log in **logpath shows** the logins **maxretry** sets how many retries to allow before ban **bantime** is in seconds and **ignoreip** ignores an ip. We now need to restart it by doing the command **sudo systemctl restart fail2ban.service** verify the status of the service by using the **sudo systemctl status fail2ban** you should see a message as seen in the picture

A computer screen with white text

Description automatically generated

We can a test run by logging in the server and putting the password I will do this in my cmd with ssh

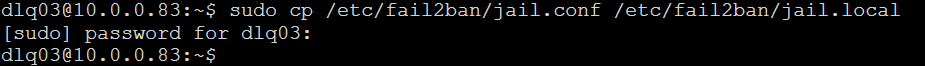
A screen shot of a computer

Description automatically generated

You stop being prompted to log in after 3 failed attempts 3 attempts in my file is the limit before a 1 hour ban logging in again after that should have it say connection timed out

Ubuntu

The process of installing fail2ban on Ubuntu is pretty much the same type **sudo adt install fail2ban** then enter password then **y** when prompted to confirm installation. We also need to create a local jail for the setup work. We can manually create a file in the file path **/etc/fail2ban/** and manually put the output or copy the contents of **/etc/fail2ban/jail.conf** which is pretty much a preset for settings but not the actual configuration. We can do this with the command **cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local** which will copy the contents of **jail.conf** into **jail.local**



Opening **jail.local** and navigating through the files show the following settings

**A screenshot of a computer

Description automatically generated**

**Bantime** shows how long a user is banned **findtime** is the timeframe a user has exceeded the **maxretry** which in this is 5 during the **findtime** frame which is 10 minutes and **maxretry** shows max attempts

After establishing a jail we need enable it with the following command **sudo systemctl enable fail2ban** and **sudo systemctl start fail2ban**

A computer screen with text

Description automatically generated

Trying to establish a connection after the amount of failed attempts should eventually lead to this when trying to connect from elsewhere



Lolcat

CentOS

make sure you the **epel-release** for you centOS serverrepository as the program is located there you can do this with the following command **sudo yum install epel-release** (Ignore if already installed)

A screenshot of a computer program

Description automatically generated

Once that is installed you need to the following commands

**sudo yum install snapd** – to install snap which is what will be used to install lolcat

**sudo systemctl enable --now snapd.socket** – which enables the way snap communitcates

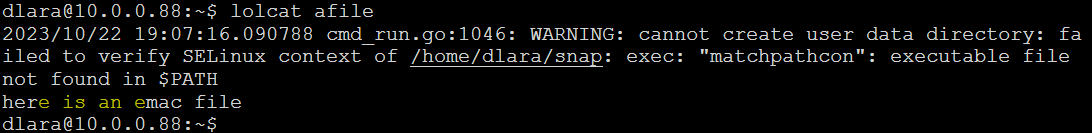
**sudo ln -s /var/lib/snapd/snap /snap – to turn on classic snap and make link between normal and classic snap**

**Once those are done install snap with the following** command **sudo snap install lolcat**

**A black screen with white text

Description automatically generated**

We can use lolcat to see a more colorful version of a file



(This is bugged and it may be because I am using CentOS 7 when this is meant for CentOS 7.6 and above)

Lolcat location

A black background with white text

Description automatically generated

Ubuntu

We can install lolcat on Ubuntu with the command **sudo apt install lolcat**



This will then prompt for password enter it and wait for the text that shows the process to finish we can test lolcat by doing **lolcat filename**

A black background with white text

Description automatically generated

Lolcat location

A black background with white text

Description automatically generated

Cowsay

CentOS 7

To install Cowsay on CentOS 7 I did the following command **sudo dnf install cowsay**

A screenshot of a computer

Description automatically generated

Location

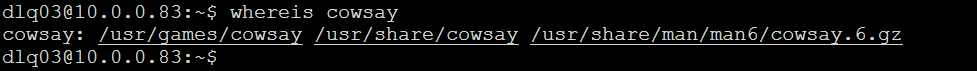


Ubuntu

The command to install cowsay on Ubuntu was **sudo apt install cowsay**A screenshot of a computer program

Description automatically generated

Location of **cowsay**



Vim

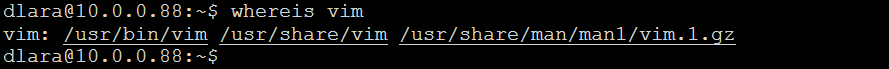
CentOS

To install Vim on CentOS simply put the command **sudo dnf install vim**

A black screen with white text

Description automatically generated

Vim Location Below



Ubuntu

To install Vim on Ubuntu simply do the following sudo command **sudo apt install vim**

A screenshot of a computer

Description automatically generated

Vim Location below



Sources

<https://ostechnix.com/install-dnf-centos-7/>

<https://web.archive.org/web/20230308044120/https://wiki.centos.org/AdditionalResources/Repositories>

<https://www.networkworld.com/article/3305810/how-to-list-repositories-on-linux.html>

<https://unix.stackexchange.com/questions/474664/how-to-install-tmux-on-centos>

<https://docs.suricata.io/en/suricata-7.0.2/install.html#ubuntu-from-personal-package-archives-ppa>

<https://askubuntu.com/questions/1415009/add-apt-respository-keeps-timing-out>

<https://linuxhint.com/emacs_save_quit/>

<https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-centos-7>

<https://www.linode.com/docs/guides/how-to-use-fail2ban-for-ssh-brute-force-protection/>

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

<https://snapcraft.io/install/lolcat/ubuntu>