**Continue Setup: CentOS**

1. **Install updates - “sudo yum update”**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* Power on the CentOS server and run this command. Yum is short for ‘Yellowdog Updater Modified’, which is used for installing and updating software packages in all Red Hat based Linux systems, like CentOS and Fedora. The system will prompt for the admin password before executing the command. It is always recommended to take a snapshot of the VM before updating, to have a recent restore point. Note when the update took place – 17OCT24.

1. **Install Tmux – “sudo yum install tmux”**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* **What is tmux?** Tmux is an abbreviation for ‘Terminal Multiplexer’. This tool allows you to create multiple terminals out of a single session. It gives the ability to run multiple programs with a single terminal, or connection, to the system.

A screenshot of a computer

Description automatically generated

* The command **tmux** by itself will open a new session with a generic name like ‘0’. **“tmux new -s Example”** will initiate a tmux session named Example. I prefer this command because it applies an actual name to the session window which is better for keeping organized.

A black and white screen with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

* Use the commands below to better utilize the advantages of tmux. Users can create new windows and navigate through them using these shortcuts. The output of ‘tmux ls’ shows how many active windows or sessions are open. At the bottom of the window highlighted in green, it shows I have the 6th window selected, the asterisk is the indicator. If we wanted to switch to the 4th window for example, I would press **Ctrl + B, P** twice to get there.

|  |  |
| --- | --- |
| Ctrl + B, C | Creates a new window. |
| Ctrl + B, N | Switch to the next window. |
| Ctrl + B, P | Switch to the previous window. |
| Ctrl + B, D | Detach from the session (session continues running in the background). |
| Ctrl + B, W | List all windows and switch between them. |
| Ctrl + B, X | Kill the current pane or window. |

1. **Install EMACS – “sudo yum install emacs”**

A screenshot of a computer

Description automatically generated

* **What is EMACS?** Emacs is a text editing tool on Linux distros that is similar to Nano but has more advanced features and customized options for users. The installation and output of the command is lengthy, be sure to select ‘Yes’ to confirm when it prompts you during the installation.

A screenshot of a computer

Description automatically generated

* The command **‘emacs AwkScript’** will open the contents of the file. Here we can make manual edits to the script, or any other text file. The configurations for most programs, like Fail2ban, are just text files. This is a great use for emacs and is an efficient way to chane programs settings. When finished, press ***CTRL + X*** followed by ***CTRL + C***to exit the emacs editing tool.

1. **Install Fail2ban – “sudo yum install fail2ban”**

* **What is Fail2ban?** Fail2ban is a useful security tool that automatically blocks IP Addresses that fail to authenticate credentials to the server. For example, if a hacker was trying to get into a server, they might use an attack which tries infinite account credentials to gain access to the server. If Fail2ban recognizes a suspicious number of attempted logins from the same IP Address, it will then block the IP.

A screenshot of a computer

Description automatically generated

* On CentOS, Fail2ban is not as straightforward to install compared to Ubuntu. The repository that contains Fail2ban is not included by default and needs to be enabled by the Administrator. **“sudo yum install epel-release”** will enable the Extra Packages for Enterprise Linux, **epel** for short, repository. Now the system is ready to install Fail2ban**.**

A screenshot of a computer

Description automatically generated

* Run the command above and enter ‘y’ when prompted to install fail2ban. The output will say ‘Complete!’ when finished.

A screenshot of a computer

Description automatically generated

* Once installed, open the configuration file for Fail2ban, the location of this config file is /etc/fail2ban/jail.conf. Copy and save it as a .local file, the read me instructs users not to edit the jail.conf file, as it is likely to be overwritten by future updates. Use emacs to open the jail.local file we just created to edit the settings. In the config file, there are a few variables we can adjust. For example, **bantime** is the number of seconds the IP Address is blocked for. I changed this to 1200 seconds (on 17OCT24) to give more time between lockouts for suspected hackers. I also set the ‘**maxretry’** variable to 3, so after 3 failed login attempts that IP will be banned. When you are finished making changes, press *CTRL + X* and *CTRL + S* to exit and save the config.

A screenshot of a computer

Description automatically generated

To set this up for SSH connections, scroll down in the Jail.local file to ‘SSH servers’, add the line **enabled = true** to activate this jail. I also changed the ‘**maxretry’** variable to 3 to block the IP Address after 3 attempted logins. Date changed – 17OCT24.

A screenshot of a computer

Description automatically generated

* To start the Fail2ban service, enter the command **“sudo systemctl start fail2ban”.** And then enable it by entering **“sudo systemctl enable fail2ban”.** Last, confirm the service is running by running a status check command – **“sudo fail2ban-client status”.** A screenshot of a computer

  Description automatically generated

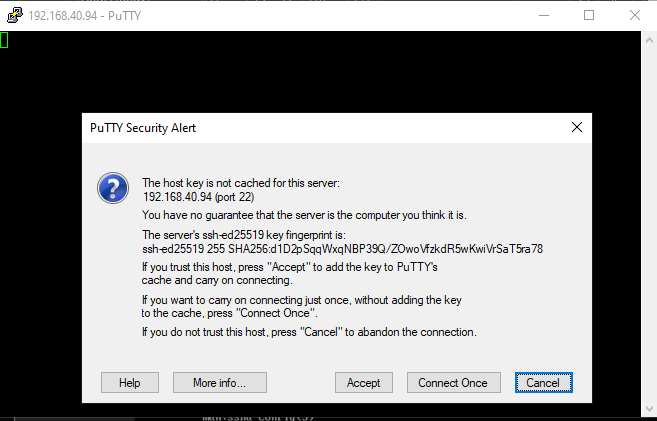
1. **Testing Fail2ban**

A screenshot of a computer

Description automatically generated

* To test Fail2ban, we need to confirm that SSH is enabled on the CentOS server. Command **“sudo yum install openssh-server”** will install the ssh feature. The output above shows ssh was already installed. Installing this tool opens the ability to connect to the server with putty or other remote applications. **“Sudo systemctl start sshd”** and **“sudo systemctl enable sshd”** starts and enables the ssh connection. Then I used **“sudo systemctl status sshd”** to show the status of the ssh service. After confirming SSH is active, we can now use Putty to test the fail2ban tool.

A screenshot of a computer

Description automatically generated  


* If it’s not already installed, download Putty ([Link Here](https://www.putty.org/)) then launch it. Using the IP Address of the server, connection type ‘SSH’, and Port 22, open the terminal session, click ‘Accept’ on the security prompt.

A computer screen with a error message

Description automatically generated

* Try to login with credentials you know are **invalid**. The point of the test it to confirm the lockout variables are working. Going back to the configuration file, we set the **‘maxretry’** variable to 3 for ssh logins. So, after 3 failed logins from the same IP Address, Fail2ban blocks all activity from that address for 20 minutes as per the ‘bantime’ variable we set earlier. Once locked out from ssh, go back to the VM and run sudo fail2ban-client status sshd to see an updated list of banned IP Addresses.

A screenshot of a computer

Description automatically generated

Above screenshot is the output of that command both before and after the IP Address block took place.

1. **Install Cowsay – “sudo yum install cowsay”**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* **What is Cowsay?** Cowsay is a fun program that takes text and outputs it as a spoken text, or bubble quote of an ASCII art Cow. If you are unfamiliar, ASCII is an art style that uses only computer text and keyboard input to draw a picture. In this case, they drew a cow using mostly back slashes (\) and other special characters. Simply use the command ‘cowsay’ and put in quotes the words you would like to cow sketch to speak! Like previous installs, select yes when prompted to confirm the installation.

1. **Install LolCat – “sudo yum install lolcat” sudo yum install ruby**

A screenshot of a computer

Description automatically generated

* **What is LolCat?** Lolcat is a command line utility that is like the command ‘cat’ (which is abbreviated for concatenate). The difference is the rainbow-colored text that lolcat outputs. Because lolcat is written in ruby programming language, I first need to install Ruby for lolcat to work. Run **“sudo yum install ruby”** to begin the install.

A screenshot of a computer

Description automatically generated

* Once the Ruby install is finished, run **“sudo gem install lolcat”.** Gem is the software package installer for all Ruby programs. We have been using ‘yum’ for this up until now.

A screenshot of a computer

Description automatically generated

* In this example, I ran the command ‘lolcat Sedlab’. This displays the contents of the SedLab just like the cat command would, but with coloring.

A screenshot of a computer

Description automatically generated

1. **Install Vim – “sudo yum install vim”**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* **What is Vim?** Vim is another text editor used in Linux environments. It is a newer version of Vi that added a more user-friendly experience that includes mouse support. We can check if Vim is installed by typing ‘Vim’ into the command line. If it is not installed, run the install command listed above to complete the installation. Use **“vim *FileName*”** to edit files.

1. **Find Files – “rpm -ql *PackageName*”**

* To find the location of a package, or installation, use **“rpm -ql list tmux”** to list all files associated with tmux. RPM -ql is telling the system to query and list all files associated with the package listed by the user.

A screenshot of a computer screen

Description automatically generated

* To see everything installed on the server, use a slightly different command – **“yum list installed”.** See below screenshot for output from this command. I would recommend directing the output of this command to a text file, because the results are lengthy, and it is much easier to sift through as a text file. So, the full command would look like **“yum list installed >> *NewFile*”.**

A screenshot of a computer

Description automatically generated

**Sources:**

[**https://superuser.com/questions/488890/list-of-installed-repositories-yum**](https://superuser.com/questions/488890/list-of-installed-repositories-yum)[**https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-centos-7**](https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-centos-7)

[**https://www.aholdengouveia.name/LinuxAdmin/installs.html**](https://www.aholdengouveia.name/LinuxAdmin/installs.html)