Enoch Masih

Linux Administration

Firewalls

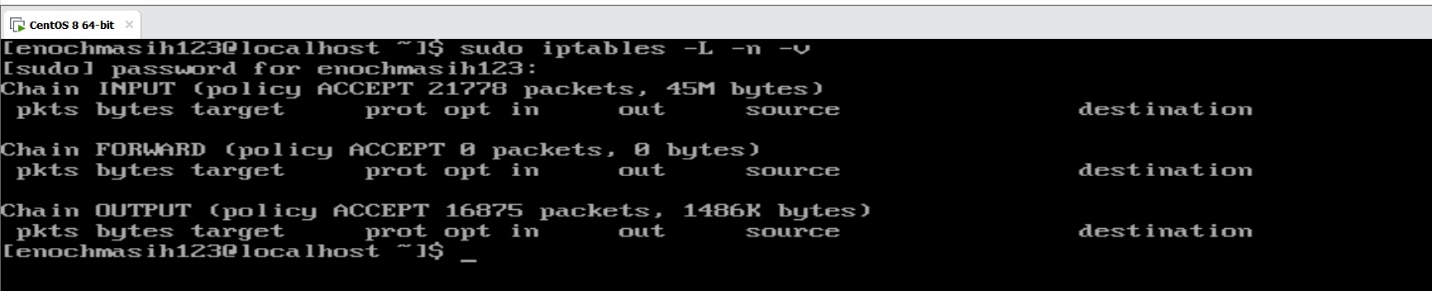
**Pre-Configurations**

Before starting the Lab I made sure that I had Wireshark, Vmware running CentOS, and Apache.

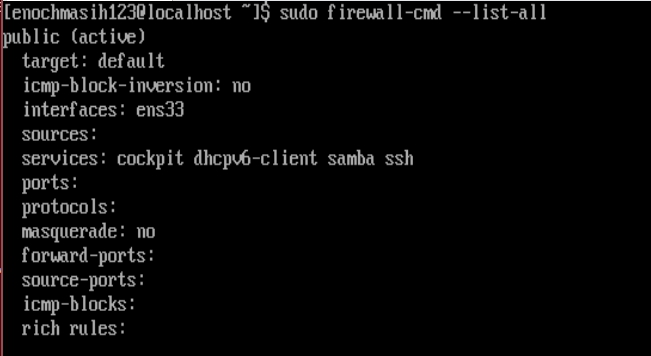
Apache can be installed using the dnf package manager. The install command is sudo dnf install httpd.

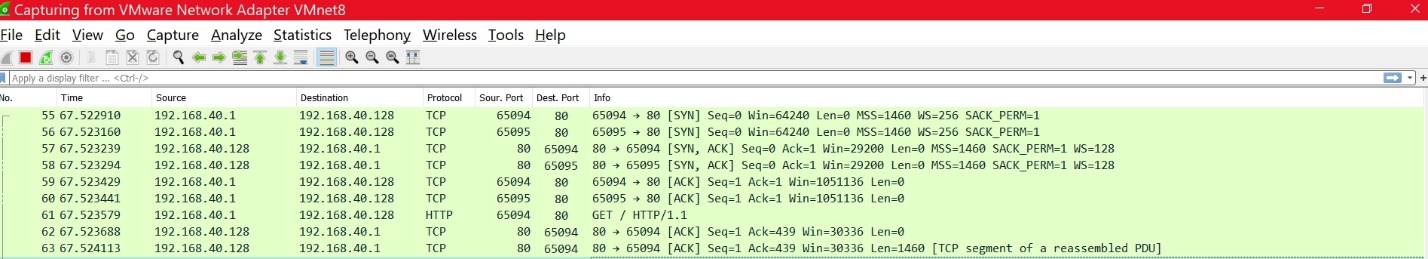
Once installed make sure to start the service using system control. the command is sudo systemctl start httpd. you can check to make sure that the service is running by using the previous command and replacing start with status. A positive value will be an active (running) result.

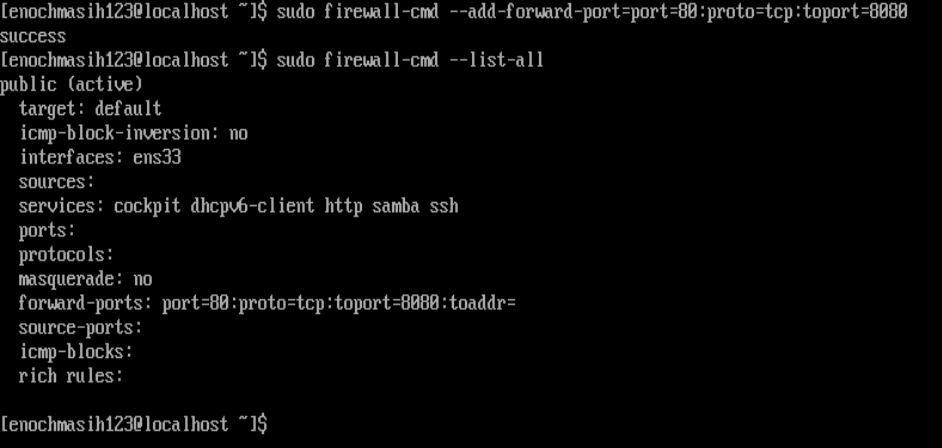
**Linux Firewall**

Linux distributions provide several different methods of implementing security. Linux approaches firewall based on tables, chains and rules. There are three basic tables: Filter, NAT, and Mangle. Filter is the most common and set as the default in the iptables command. Chains indicate which part of the journey the packet is in. Input and output are two common stops in a packet journey. During these ‘stops’ one can implement rules. Rules specify what to allow forward in the chain journey or what to stop. There are two ways to stop; DROP or REJECT. Drop will simply drop the packet and move on. While reject will inform the source IP that the packet was filtered out. Below is my original Ip table in CentOS. I used the super user do command to access the iptables command. In iptables, I used some options to specify the result. I used the -L option to list out the table, -n to show the numeric format of ip addresses instead of the default hostnames and -v to show verbose which prints out computational messages. By default it is showing the filter table which has three chains Input, output, and forward. 

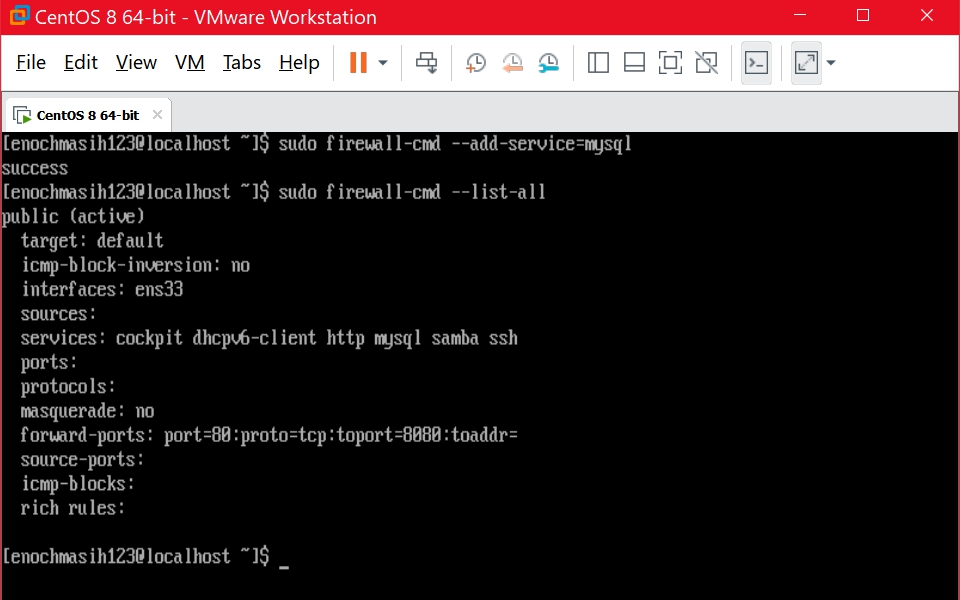
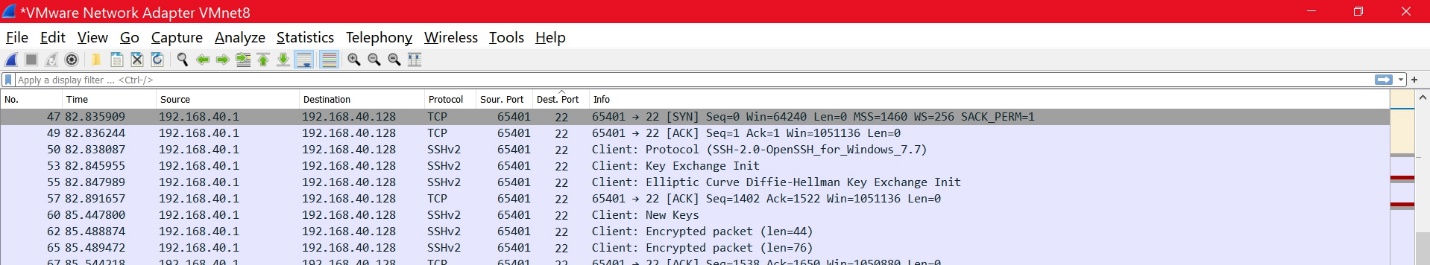
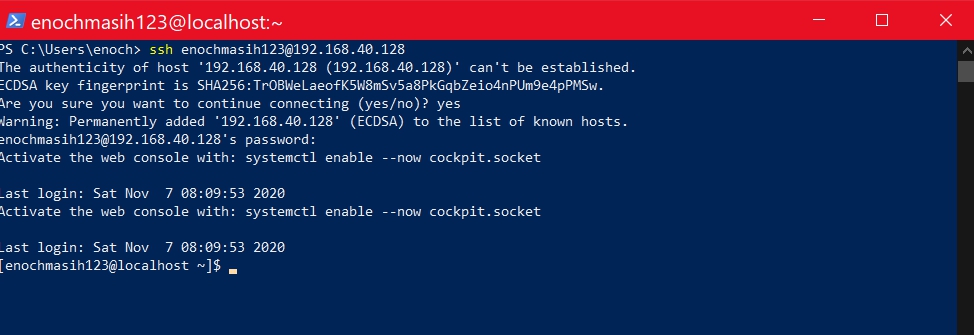
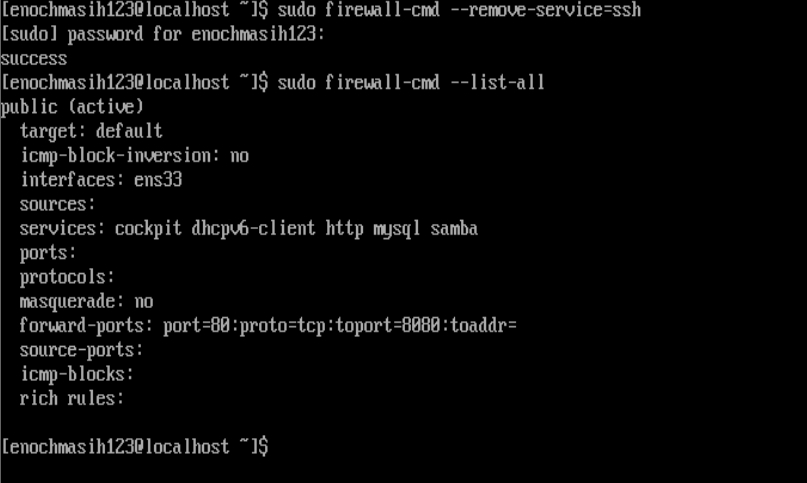
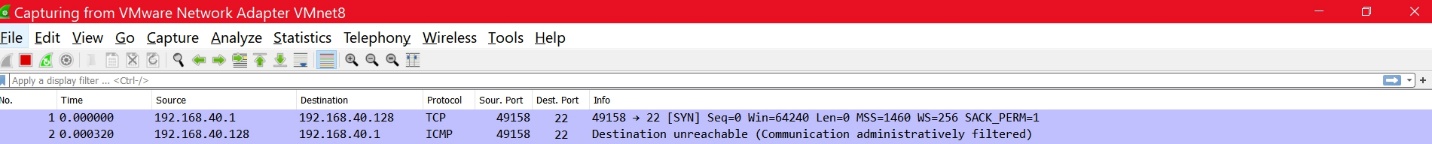
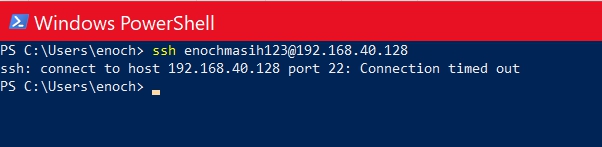
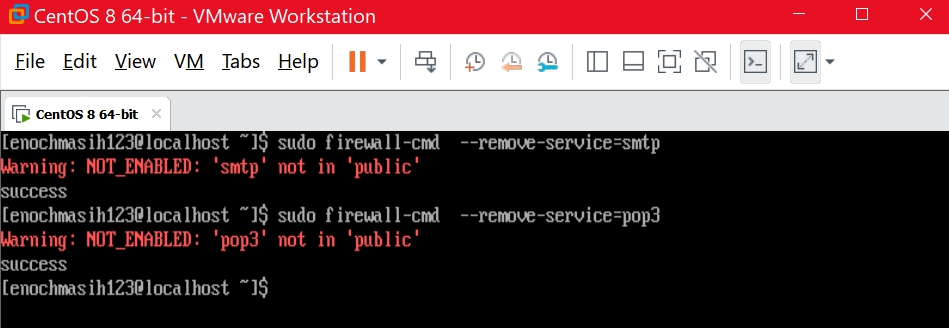
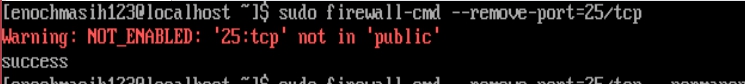
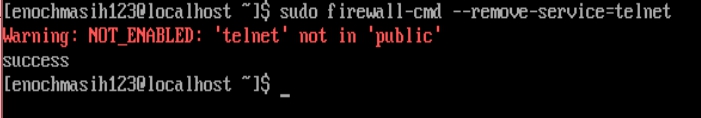
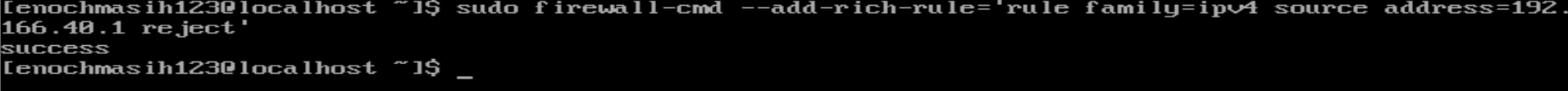
One can install another utility named firewalld. In the CentOS environment, this came pre installed. Make sure to update all services before conitnuing. In addition to the features offered in iptables, firewalld also offers the addition of services which may simplify proccess as well as zones. One way to check which services are currently running is to use the following command. Note that http is currently

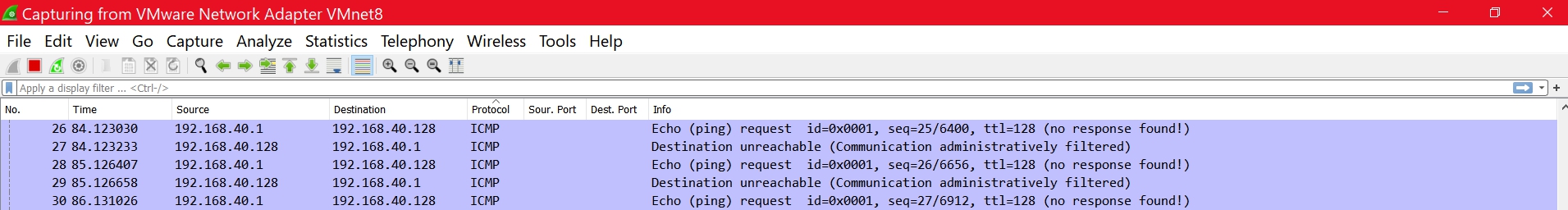
not listed as service that is running. 

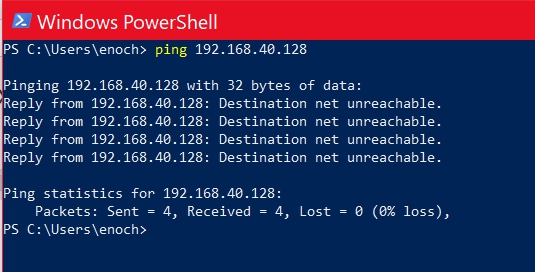
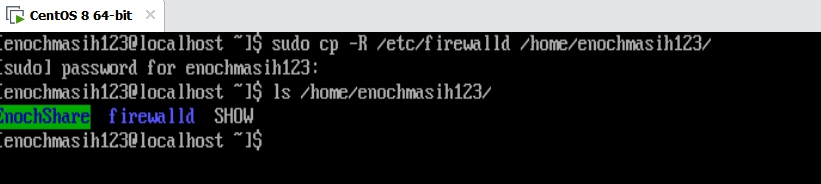
1. Below is a screenshot of opening the HTTP which by default uses port 80. A positive result is indicated by displaying the message ‘success’. I then checked the services that are currently operational by listing all services, which now includes http. Another proof is to run Wireshark while browsing the http protocol from another local host and observing successful packet delivery as shown in the screenshot below 

The next part of the question asks to forward port 80 traffic into port 8080. This can be done with the following command 

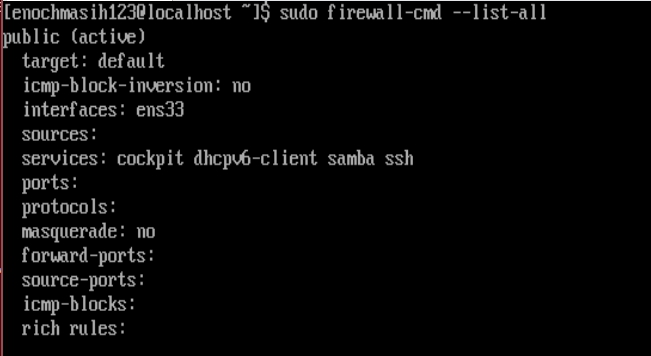
The list all command displays forward-ports. This now includes forward port 80 to 8080.

1. Mysql uses the port 3306. I opened this port by adding this service to my server as picutred below. I also checked if the service was now added by listing all the services. 
2. The SSH port 22 was already open on my Linux Server. I connected using my windows machine that is in my local network using PowerShell. Wireshark application was used to capture packet between the local host and my linux server (192.168.40.128). As displayed below the destinantion port is 22 which is the default SSH port. This shows successful communication. I used powershell to connect to linux server as shown below. In order to block SSH one can remove the service using a similar command as displayed below Now when attempting to access the server using the SSH protocol we see the following result And on PowerShell we see a similar message of the connection ‘timed out’ 
3. I choose to block Email protocols such as smtp (simple mail transport protocol) or pop3 as I would never be using it in the linux environment. As it turns out, These services are not enabled. I can also choose to specifically block the port as well as shown below. The modern day SMPT uses various ports, so as an example I used port 25 tcp to be blocked. I received a expected response from the machine.
4. Below I have listed all commands that can be used to block the listed connections
   1. to block telnet use the following command 
   2. to block a specific host use the following command using the host ip address. The rich rule option allows us to add multiple different sets of scenarios as per the need.
   3. As similar to 5b. Blocking a mac address can be done with the same option but with the source mac instead as seen below. I choose to not include a real Mac address in this example. 
   4. To Block all pings use the following command Note alternatively add request as an option after echo. 

This command can also be checked using Wireshark and PwerShell. 

1. The Directory for the configurations that were made using the firewalld command are saved in the /etc/firewalld directory. One simple way is to copy the directory into a backup folder. Make sure to use the -R option so that the directories within the firewall get copied as well therefore saving the entire directory. The screenshot for this is below which shows the copy and then the folder that is present in the home directory. 

**Before/After**

Before adding any configurations to the firewall: 

After adding the firewall rules this is what my firewall looked like at the conclusion: 

**Ddos Attack**

One primary use of a firewall is to prevent unwanted and harmful traffic that can damage the computer. A common attack used to be ddos attack where the server is bombarded with various requests that overwhelm the network. Ddos stands for distributed denial of service attack. This causes authentic and important traffic to not be able to reach the destination effectively taking the network out of use. One simple way of preventing this attack is to limit how many request the computer should process. This limitiation allows the computer to never get overwhelmed with network traffic requests. Firewalls can help create rules which can be a script used to implement this into a server. Below are two sources that can help set this within the network. The first source shows the implementation of a firewall rule against ddos attack. The other source shows the theory behind what happens in a ddos attack and what are some the signs to look out for.

**Sources for DDOS**

1. [**https://www.unixmen.com/iptables-vs-firewalld/**](https://www.unixmen.com/iptables-vs-firewalld/)
2. **https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/**