EN4720: Security in Cyber-Physical Systems
Exercise — Infrastructure Security

Name: Thalagala B. P.
Index No: 180631J

June 17, 2023

This is an individual exercise!
Due Date: 20 June 2023 by 11.59 PM

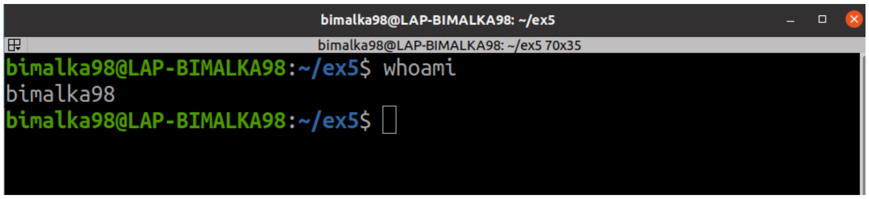
This exercise has to be carried out using a Linux-based PC/virtual machine. Read all the instructions and questions before attempting the exercise. Add answers under each question and submit the resulting PDF.

Section 1

In this section, you will implement Firewall rules using **iptables** and **ufw** Linux commands. Moreover, you will scan network ports of a remote device using **nmap** Linux command.

For all the questions in this section, add a screenshot of the terminal (including all the commands you ran to perform the task) unless specified otherwise. The evaluator should be able to see each step that you followed to perform each task. In all screenshots, the areas marked (which are unique to your terminal display) in Figure 1 (the sample answer to Question 1) must be visible.

1. View the currently logged in user.



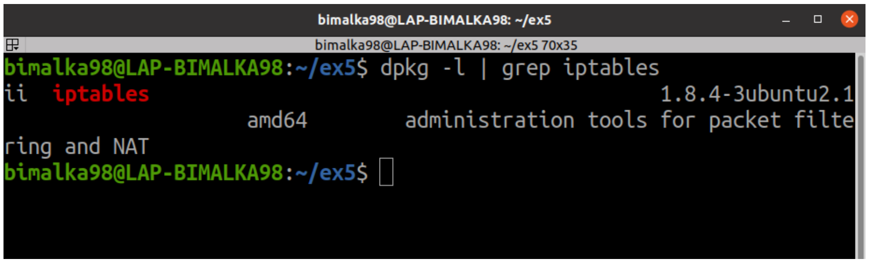
bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 70x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ whoami
bimalka98
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 1: Currently logged-in user

Creating Firewall Rules with iptables

2. Use `dpkg -l | grep iptables` command to check whether iptables is installed on your system. If it does not existing in your system, install it by running `sudo apt-get install iptables`.

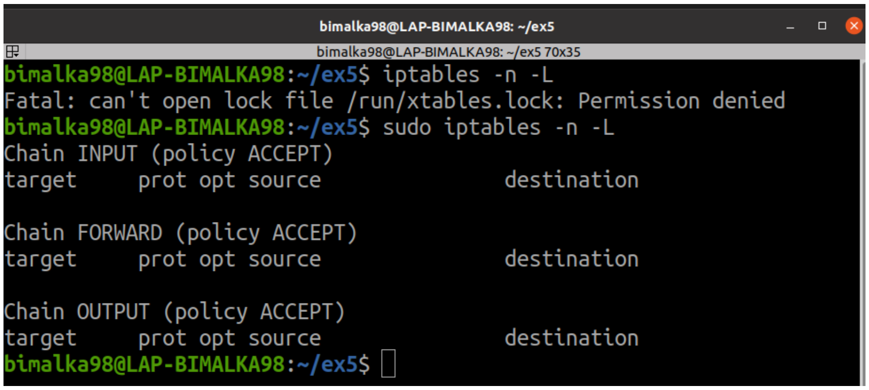


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 70x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ dpkg -l | grep iptables
ii  iptables                                         1.8.4-3ubuntu2.1
                    amd64        administration tools for packet filte
ring and NAT
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 2: Checking whether `iptables` is installed on the system

3. Check all available iptables rules in your system using the command `/sbin/iptables -n -L` .

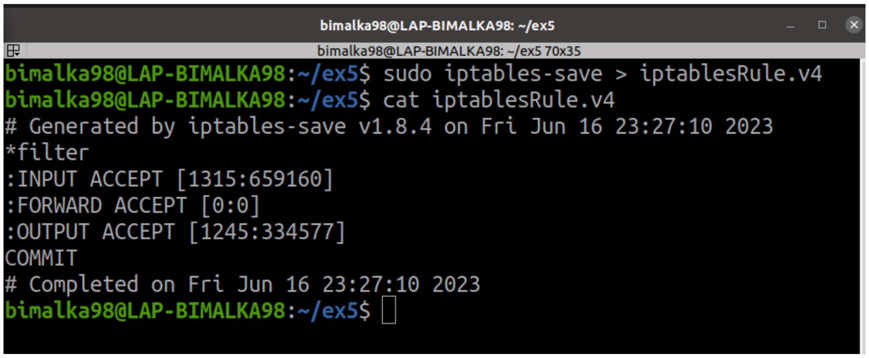


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 70x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ iptables -n -L
Fatal: can't open lock file /run/xtables.lock: Permission denied
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 3: Checking all available `iptables` rules

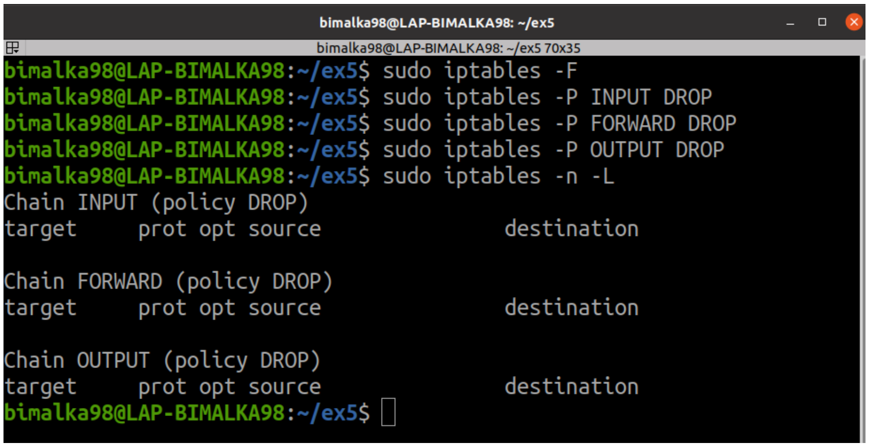
4. Save all available iptables rules to a file named **iptablesRule.v4** using `iptables-save` command.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 70x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables-save > iptablesRule.v4
bimalka98@LAP-BIMALKA98:~/ex5$ cat iptablesRule.v4
# Generated by iptables-save v1.8.4 on Fri Jun 16 23:27:10 2023
*filter
:INPUT ACCEPT [1315:659160]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [1245:334577]
COMMIT
# Completed on Fri Jun 16 23:27:10 2023
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 4: Saving all available `iptables` rules

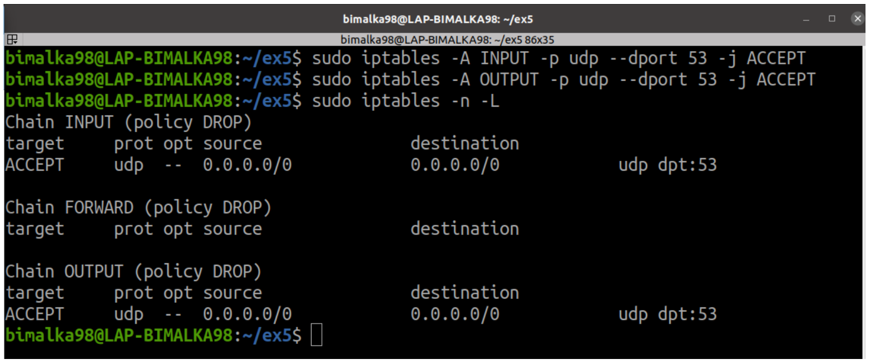
5. Flush all the iptables rules that exist in your system and set a default policy to drop packets.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 70x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -F
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -P INPUT DROP
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -P FORWARD DROP
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -P OUTPUT DROP
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 5: Flushing all the rules and setting a default policy to drop packets

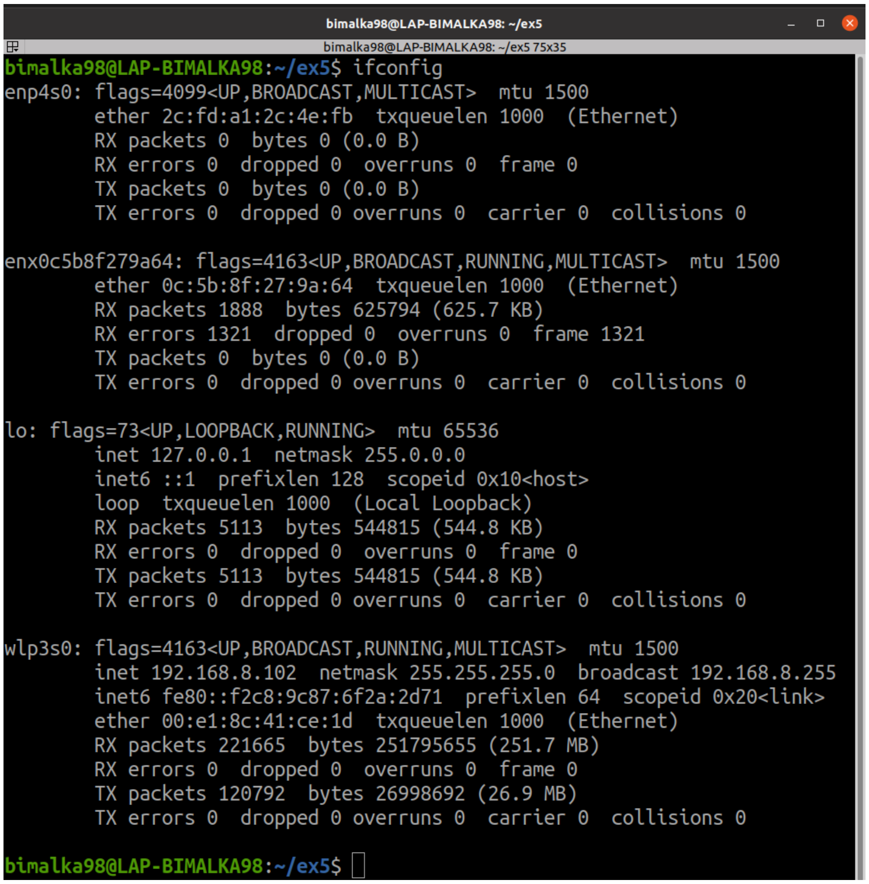
6. Set iptables rules to permit input and output DNS traffic in your system.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 86x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A INPUT -p udp --dport 53 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

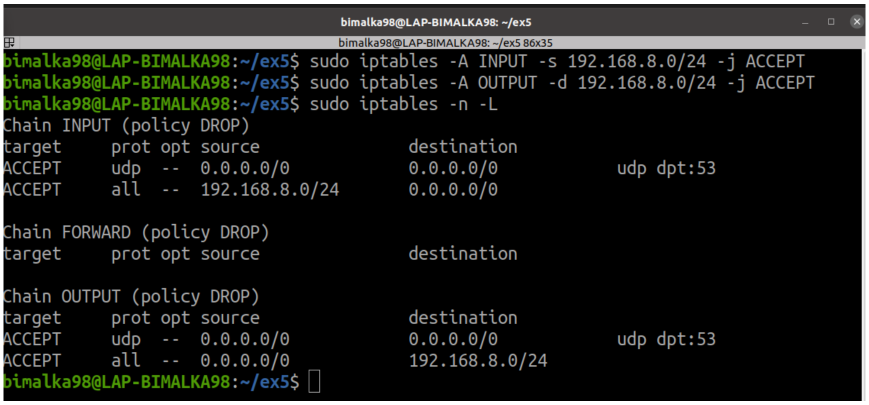
Figure 6: Permitting input and output DNS traffic

7. Add iptables rules to accept local network incoming and outgoing traffic from the network 192.168.1.0/24.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 75x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ ifconfig
enp4s0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 2c:fd:a1:2c:4e:fb  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enx0c5b8f279a64: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        ether 0c:5b:8f:27:9a:64  txqueuelen 1000  (Ethernet)
        RX packets 1888  bytes 625794 (625.7 KB)
        RX errors 1321  dropped 0  overruns 0  frame 1321
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 5113  bytes 544815 (544.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5113  bytes 544815 (544.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.8.102  netmask 255.255.255.0  broadcast 192.168.8.255
        inet6 fe80::f2c8:9c87:6f2a:2d71  prefixlen 64  scopeid 0x20<link>
        ether 00:e1:8c:41:ce:1d  txqueuelen 1000  (Ethernet)
        RX packets 221665  bytes 251795655 (251.7 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 120792  bytes 26998692 (26.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

bimalka98@LAP-BIMALKA98:~/ex5$ 
```

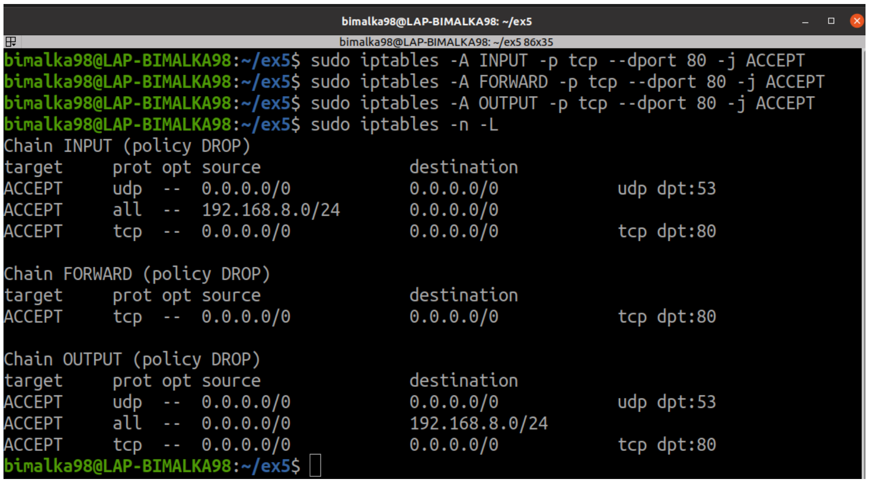
Figure 7: Checking the CIDR of the current local network


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 86x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A INPUT -s 192.168.8.0/24 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A OUTPUT -d 192.168.8.0/24 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  192.168.8.0/24       0.0.0.0/0

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  0.0.0.0/0            192.168.8.0/24
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 8: Adding rules to accept local network incoming and outgoing traffic

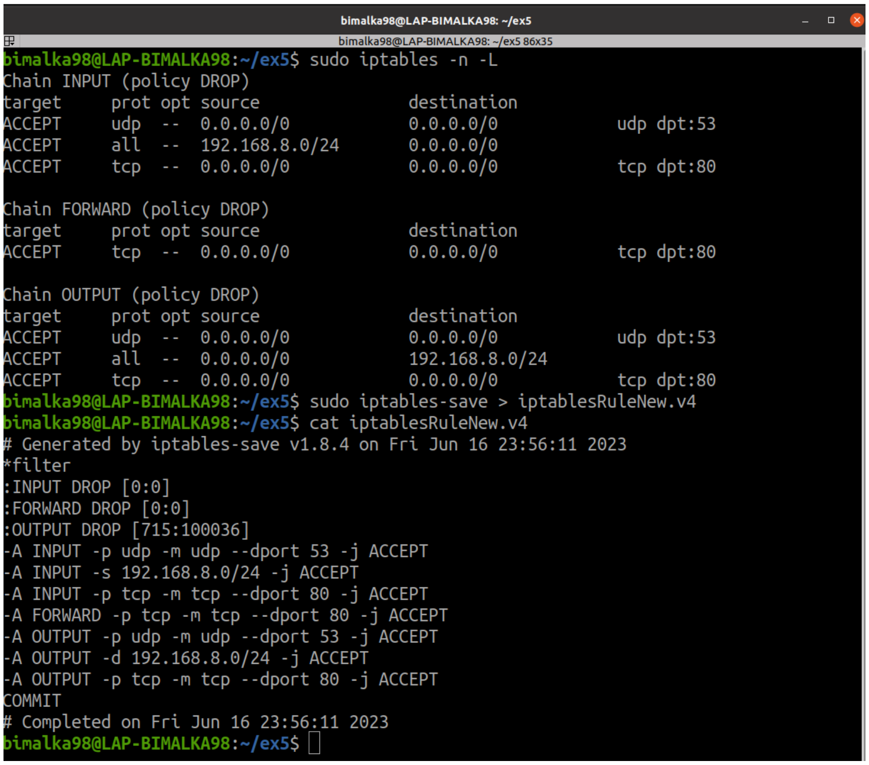
8. Configure iptables rules to allow all HTTP traffic.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 86x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A FORWARD -p tcp --dport 80 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  192.168.8.0/24       0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  0.0.0.0/0            192.168.8.0/24
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 9: Configuring rules to allow all HTTP traffic

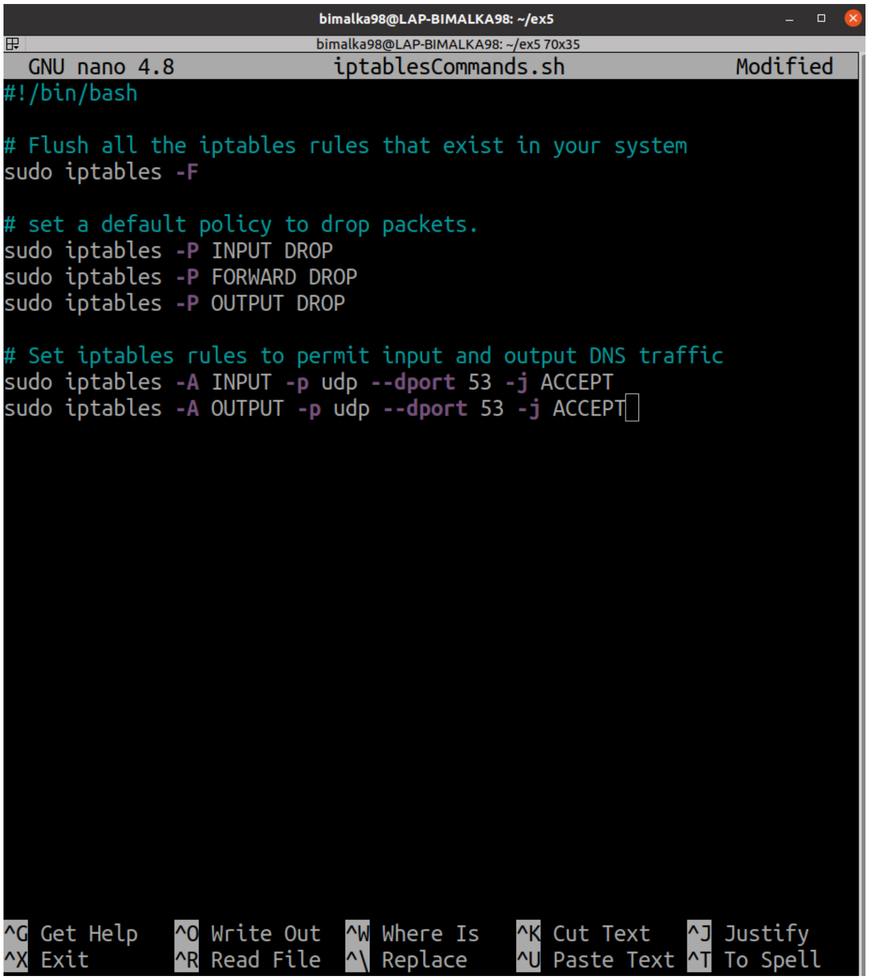
9. View all iptables rules in your system and save them to a file **iptablesRuleNew.v4**.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 86x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  192.168.8.0/24       0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  0.0.0.0/0            192.168.8.0/24
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables-save > iptablesRuleNew.v4
bimalka98@LAP-BIMALKA98:~/ex5$ cat iptablesRuleNew.v4
# Generated by iptables-save v1.8.4 on Fri Jun 16 23:56:11 2023
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT DROP [715:100036]
-A INPUT -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -s 192.168.8.0/24 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A FORWARD -p tcp -m tcp --dport 80 -j ACCEPT
-A OUTPUT -p udp -m udp --dport 53 -j ACCEPT
-A OUTPUT -d 192.168.8.0/24 -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 80 -j ACCEPT
COMMIT
# Completed on Fri Jun 16 23:56:11 2023
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

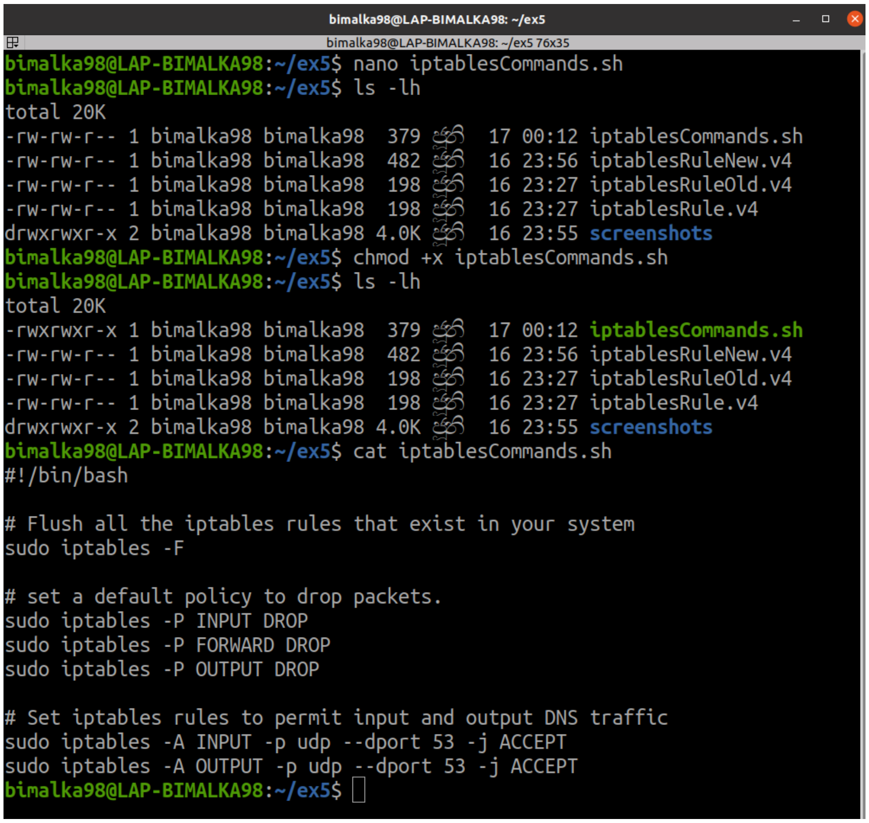
Figure 10: Viewing all rules and saving them to a file

10. Create a file called **iptablesCommands.sh** and put all commands you ran from steps 4, 5 and 6 in the file. After creating the file, flush your iptables commands again and run **iptablesCommands.sh** file. View the iptables rules now and compare with the previous result.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 70x35
GNU nano 4.8
iptablesCommands.sh
Modified

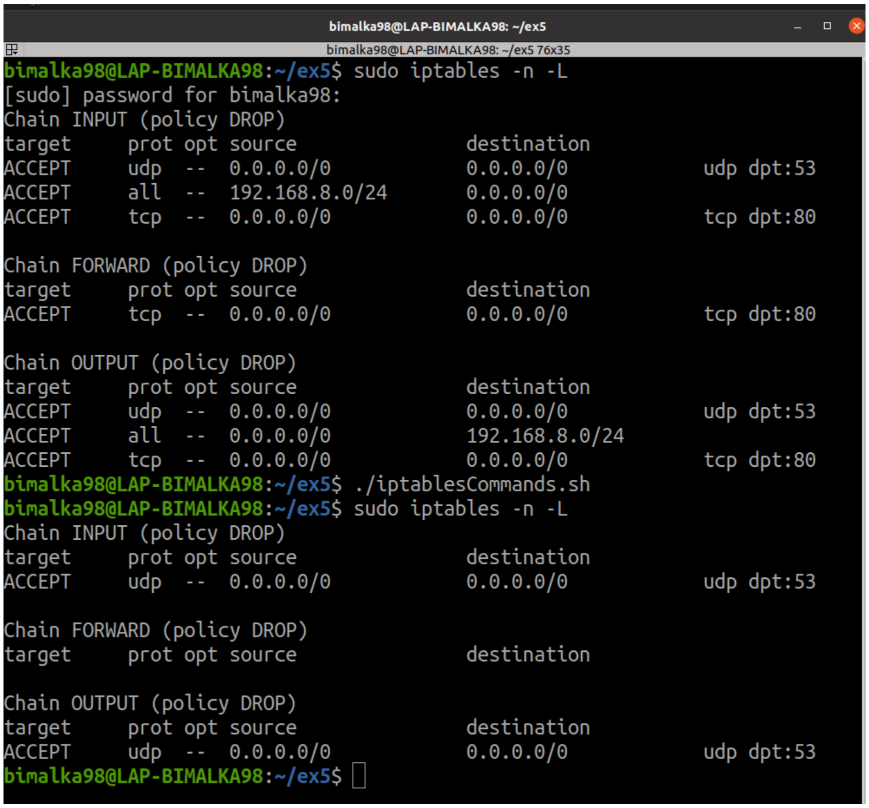
```
#!/bin/bash

# Flush all the iptables rules that exist in your system
sudo iptables -F

# set a default policy to drop packets.
sudo iptables -P INPUT DROP
sudo iptables -P FORWARD DROP
sudo iptables -P OUTPUT DROP

# Set iptables rules to permit input and output DNS traffic
sudo iptables -A INPUT -p udp --dport 53 -j ACCEPT
sudo iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
```

Figure 11: Creating the shell script


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 76x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ nano iptablesCommands.sh
bimalka98@LAP-BIMALKA98:~/ex5$ ls -lh
total 20K
-rw-rw-r-- 1 bimalka98 bimalka98  379 ජූලි  17 00:12 iptablesCommands.sh
-rw-rw-r-- 1 bimalka98 bimalka98  482 ජූලි  16 23:56 iptablesRuleNew.v4
-rw-rw-r-- 1 bimalka98 bimalka98  198 ජූලි  16 23:27 iptablesRuleOld.v4
-rw-rw-r-- 1 bimalka98 bimalka98  198 ජූලි  16 23:27 iptablesRule.v4
drwxrwxr-x 2 bimalka98 bimalka98 4.0K ජූලි  16 23:55 screenshots
bimalka98@LAP-BIMALKA98:~/ex5$ chmod +x iptablesCommands.sh
bimalka98@LAP-BIMALKA98:~/ex5$ ls -lh
total 20K
-rwxrwxr-x 1 bimalka98 bimalka98  379 ජූලි  17 00:12 iptablesCommands.sh
-rw-rw-r-- 1 bimalka98 bimalka98  482 ජූලි  16 23:56 iptablesRuleNew.v4
-rw-rw-r-- 1 bimalka98 bimalka98  198 ජූලි  16 23:27 iptablesRuleOld.v4
-rw-rw-r-- 1 bimalka98 bimalka98  198 ජූලි  16 23:27 iptablesRule.v4
drwxrwxr-x 2 bimalka98 bimalka98 4.0K ජූලි  16 23:55 screenshots
bimalka98@LAP-BIMALKA98:~/ex5$ cat iptablesCommands.sh
#!/bin/bash

# Flush all the iptables rules that exist in your system
sudo iptables -F

# set a default policy to drop packets.
sudo iptables -P INPUT DROP
sudo iptables -P FORWARD DROP
sudo iptables -P OUTPUT DROP

# Set iptables rules to permit input and output DNS traffic
sudo iptables -A INPUT -p udp --dport 53 -j ACCEPT
sudo iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

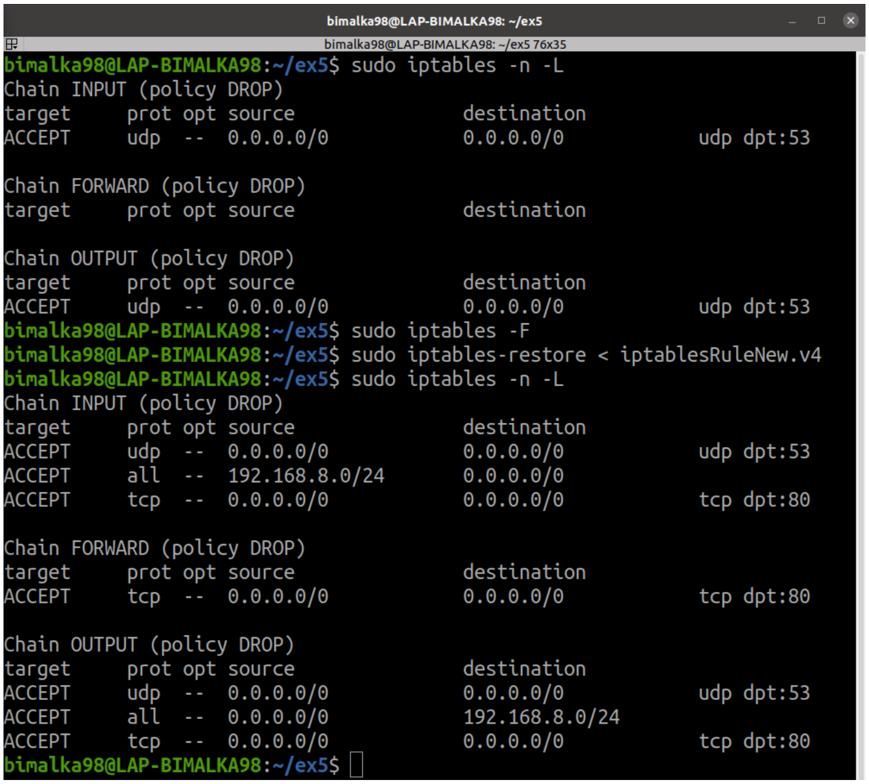
Figure 12: Making the script executable


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 76x35

```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
[sudo] password for bimalka98:
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  192.168.8.0/24       0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  0.0.0.0/0            192.168.8.0/24
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
bimalka98@LAP-BIMALKA98:~/ex5$ ./iptablesCommands.sh
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 13: Executing the shell script

11. Finally, flush your iptables rules again. But this time, load the saved iptables rules from the file **iptablesRuleNew.v4** using `iptables-restore` command. View the iptables rules and compare them with the ones you have in step 8.


bimalka98@LAP-BIMALKA98: ~/ex5
bimalka98@LAP-BIMALKA98: ~/ex5 76x35

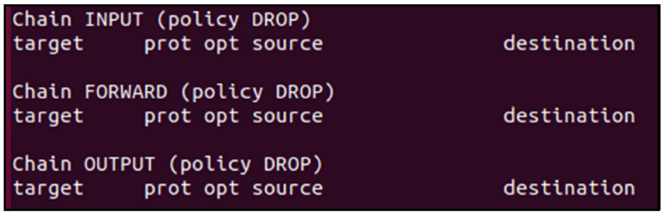
```
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -F
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables-restore < iptablesRuleNew.v4
bimalka98@LAP-BIMALKA98:~/ex5$ sudo iptables -n -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  192.168.8.0/24       0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp dpt:53
ACCEPT     all  --  0.0.0.0/0            192.168.8.0/24
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
bimalka98@LAP-BIMALKA98:~/ex5$ 
```

Figure 14: Flushing current rules and restoring saved rules

Creating Firewall Rules with UFW

The scenario comprises of two virtual machines (VM1 IP - 192.168.46.140 and VM2 IP - 192.168.46.141) running on a host (HOST IP - 192.168.46.1) machine. VM1 is an Ubuntu virtual machine that has a firewall implemented/configured.

The current firewall ruleset is as below.



```
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
```

All chain policies are set to drop traffic. To implement base rules, you can use the following commands:

- Delete any current rules associated with UFW using `sudo ufw reset`

- Disable UFW using `sudo ufw disable`
- Flush all iptables rules using `sudo iptables -F`
- Enable UFW using `sudo ufw enable`
- Deny outgoing traffic using `sudo ufw default deny outgoing`

12. Implement the following network administration in VM1:

    - Access to VM1 from VM2 must only be allowed over FTP and Telnet.
    - Access to VM1 from HOST must only be allowed over SSH
    - Allow all outgoing traffic from VM1 with the exception of access to HTTP websites

    In this task, you are asked to implement UFW rules on the ubuntu machine. You can pretend that VM2 and HOST exist in your network. List the commands you used to achieve the above. Add a screenshot of the terminal output after running the command `sudo ufw status numbered`.

    If the firewall is physically implemented, you could have tested the connections using PuTTY or the command line.

    Your answer here

Scan systems with NMAP

In this section, you will scan for the Ports of a remote host. You will need to have two devices connected to the same local network to perform this task.

13. View ip addresses of both devices using `hostname -I` command.

    Your answer here

14. Scan one host from the other host for TCP and UDP ports using `nmap` command.

    Your answer here

Section 2

15. Briefly explain VLANs, VPNs, DMZs and Network Segmentation concepts outlining their similarities and differences.

    Your answer here

16. Perform a comparison between IPsec and SSL.

Table 1: Comparison of IPsec and SSL

IPsec	**SSL**
Your answer here	Your answer here
Your answer here	Your answer here

17. Explain the differences between an IDS, an IPS, and a firewall?

    Your answer here

18. What is the difference between anomaly detection and signature or heuristic-based intrusion detection?

    Your answer here