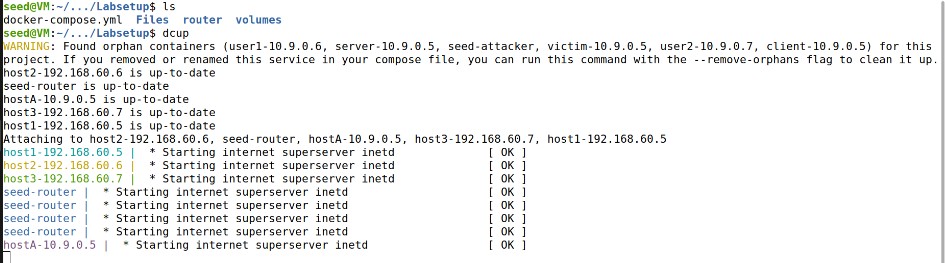
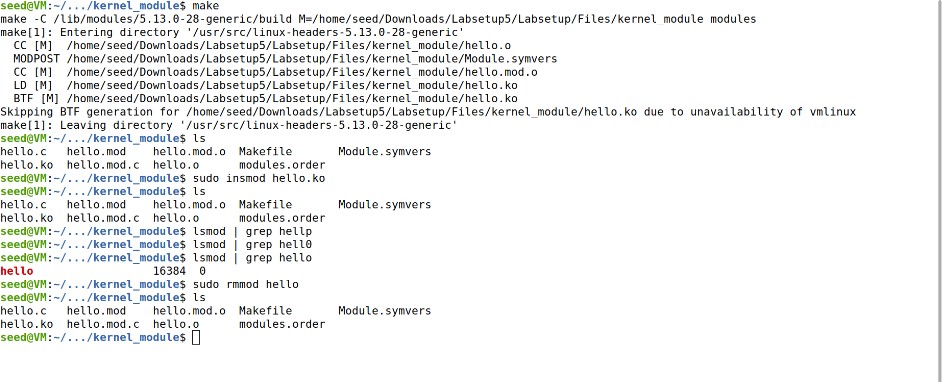
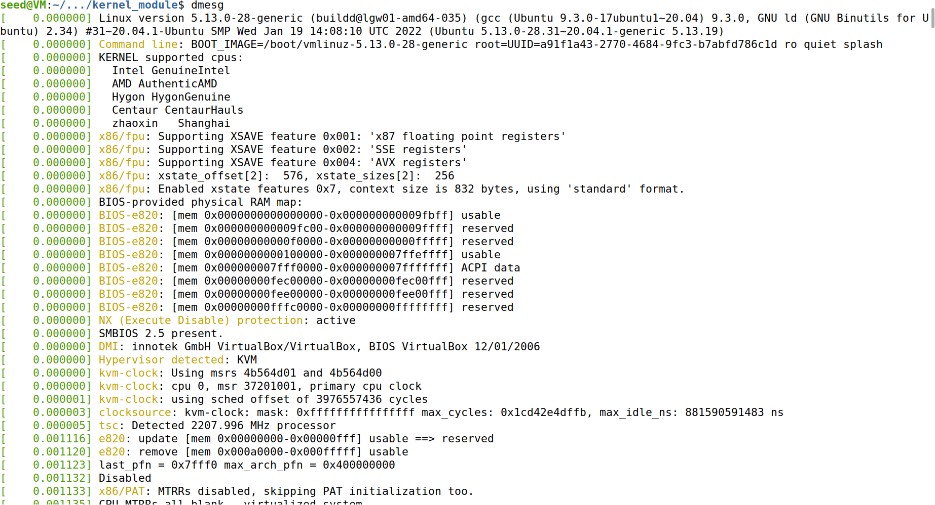
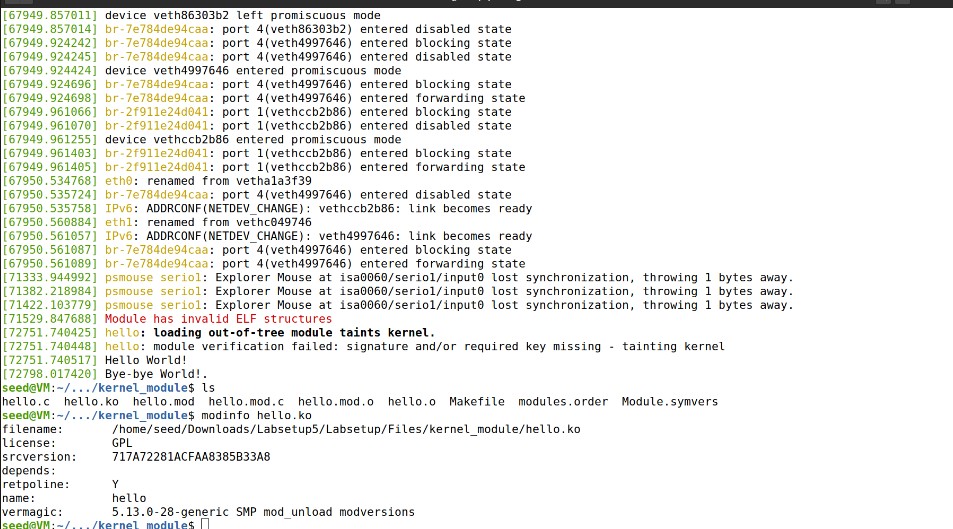
## Lab 5: Linux Firewall Exploration Lab

Task 1: Implementing a Simple Firewall

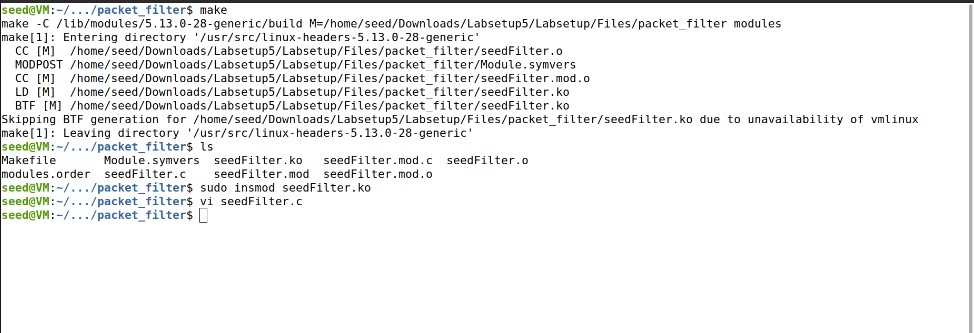
Task 1.A: Implement a Simple Kernel Module

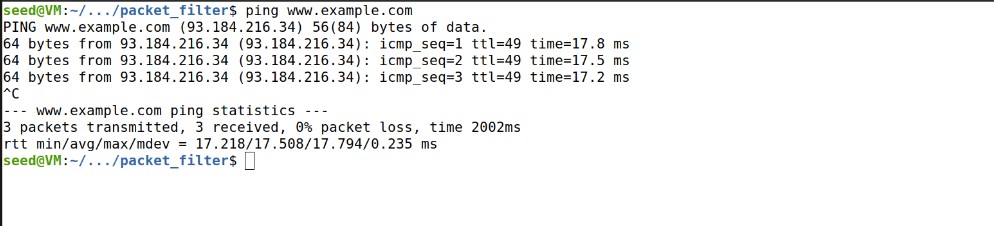
Compiling Makefile using ‘make’ command: Running following command on generated kernel module hello.ko

Executing: dmesg

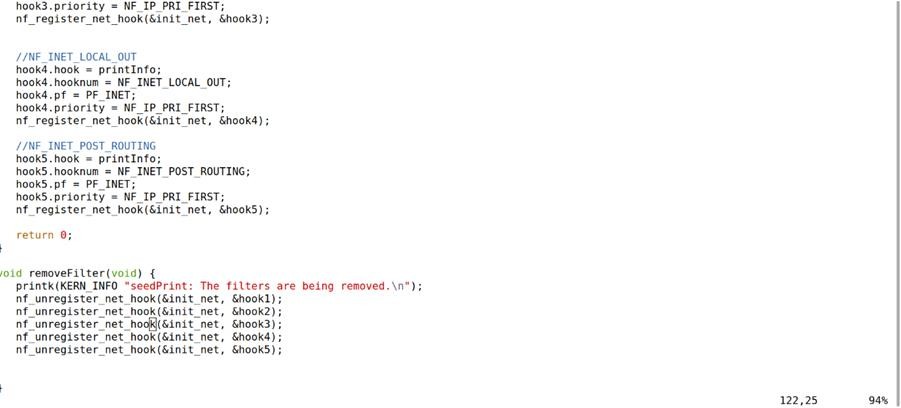
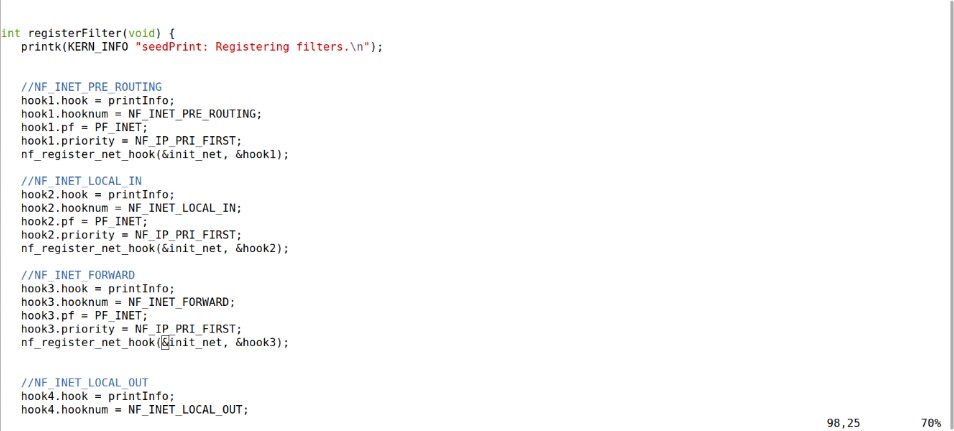
Output:

Modinfo hello.ko

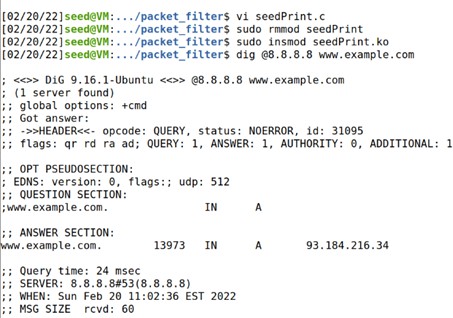
Task 1.B: Implement a Simple Firewall Using Netfilter

Checking firewall with dig @8.8.8.8 [www.example.com](http://www.example.com/)

# Task-2: Hooking printInfo function to all netfilter hooks in seedPrint.c file

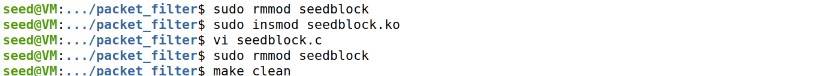
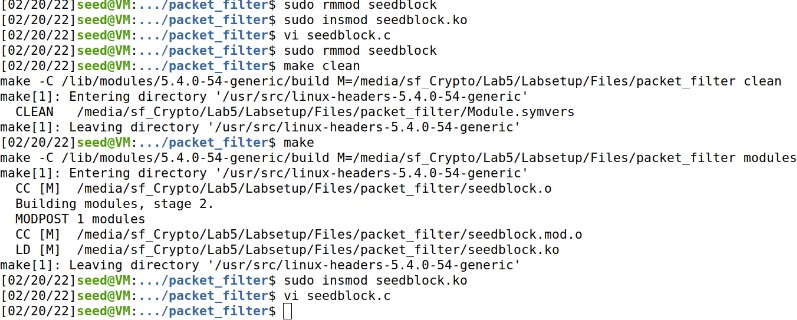


Checking the function using the command: dig @8.8.8.8 [www.example.com](http://www.example.com/)



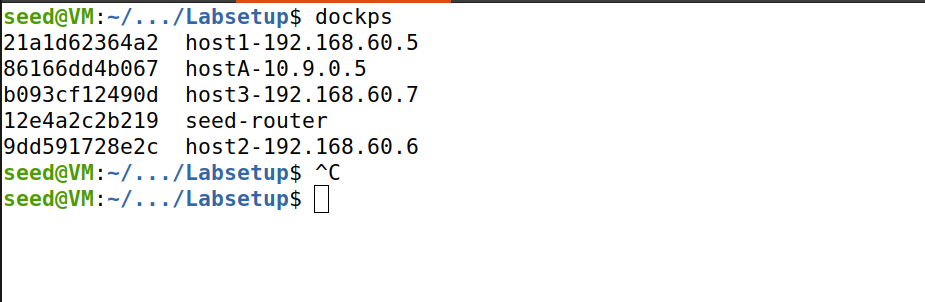
# A screenshot of a computer Description automatically generatedDmesg:

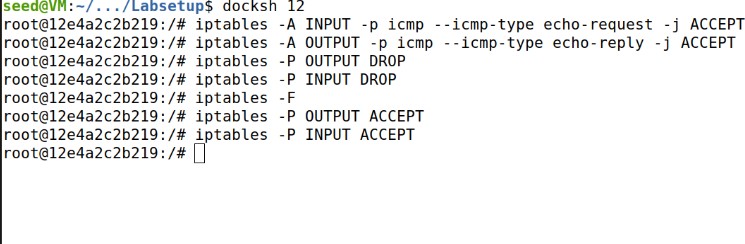
Task-3:Including hook function for blocking ping and telnet from 10.9.0.1

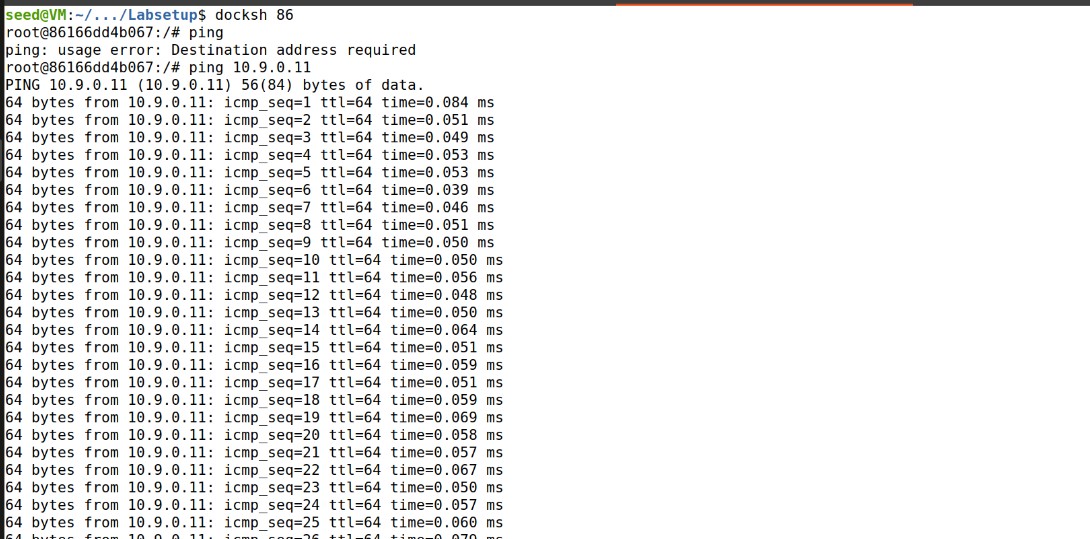


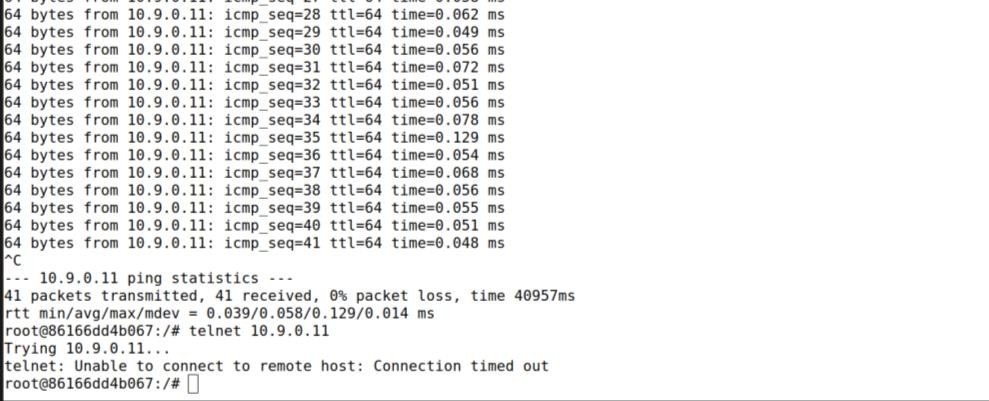
Loading seedblock.ko file to kernel using make, sudo insmod seedblock.ko command

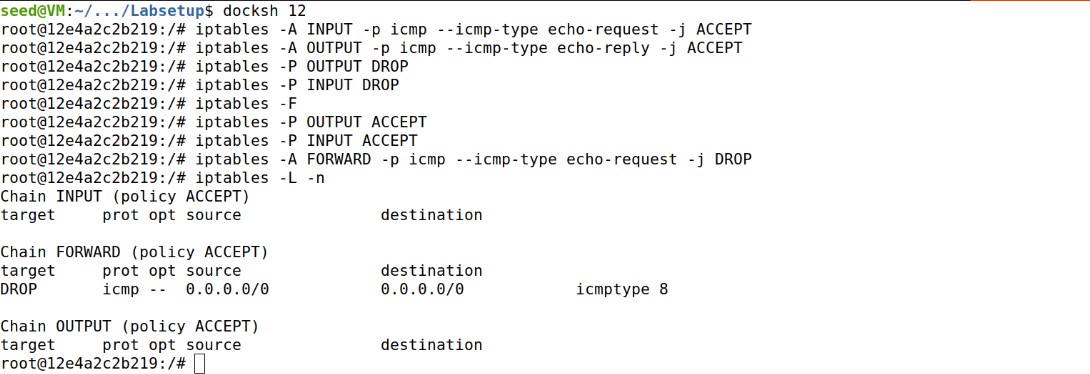
# Text Description automatically generated with medium confidenceDmesg:

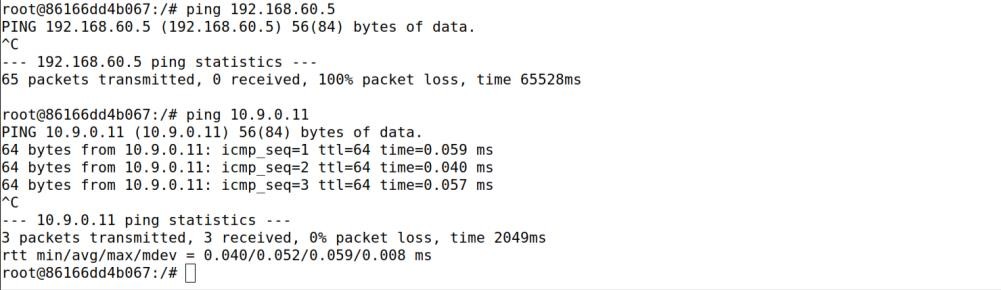
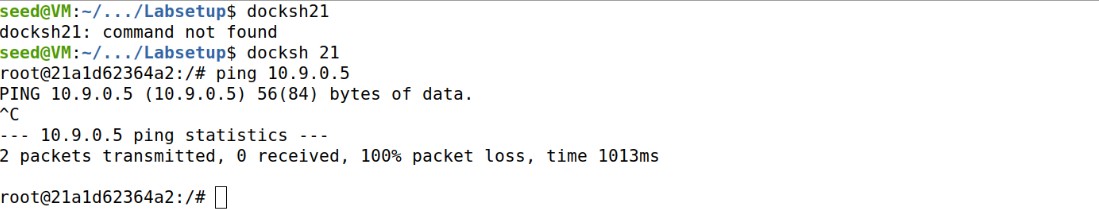
Task 2: Experimenting with Stateless Firewall Rules

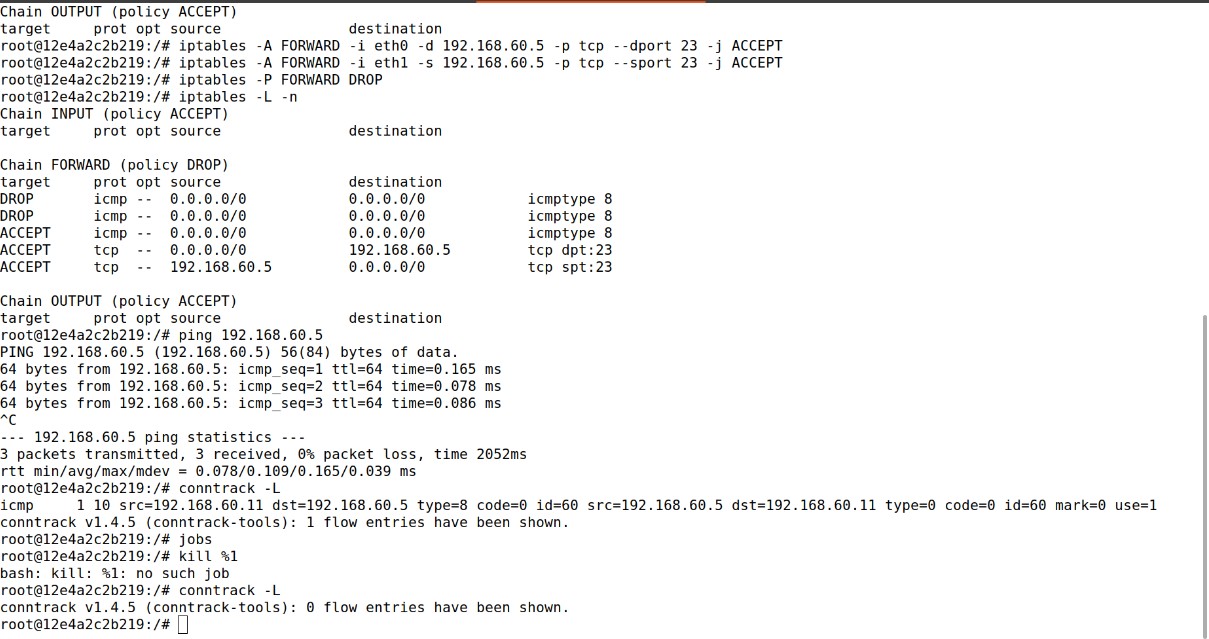
Task 2.A: Protecting the Router

Loggin to Host 10.9.0.5 using docksh command and trying to ping

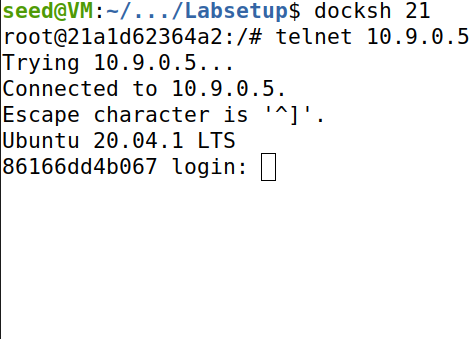
Telnet to router 10.9.0.11

Task 2.B: Protecting the Internal Network

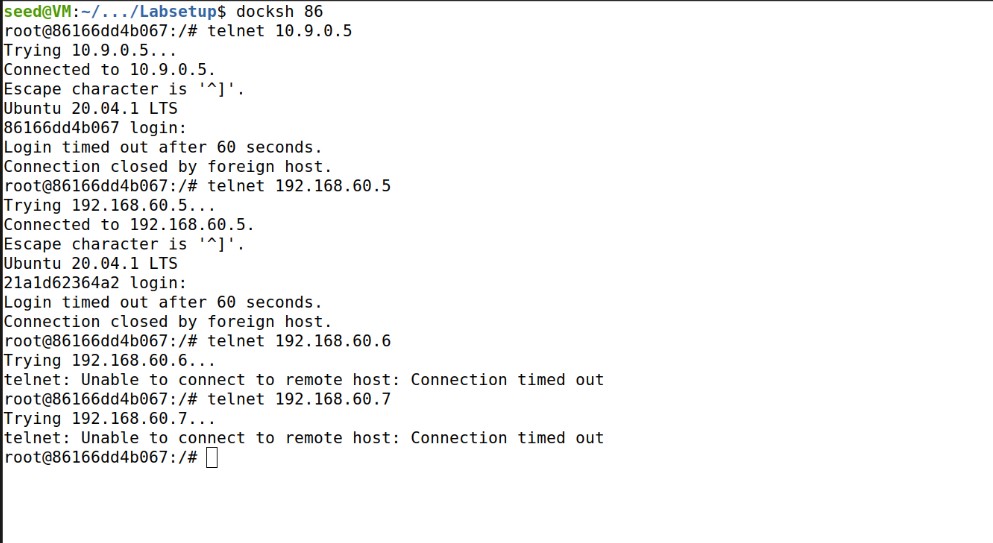
1. Pinging internal network from Host 10.9.0.5
2. Outside hosts can also ping the router.
3. Internal hosts can ping outside hosts
4. All other packets between the

internal and external networks should be blocked.

Setting up rules on router using following commands:

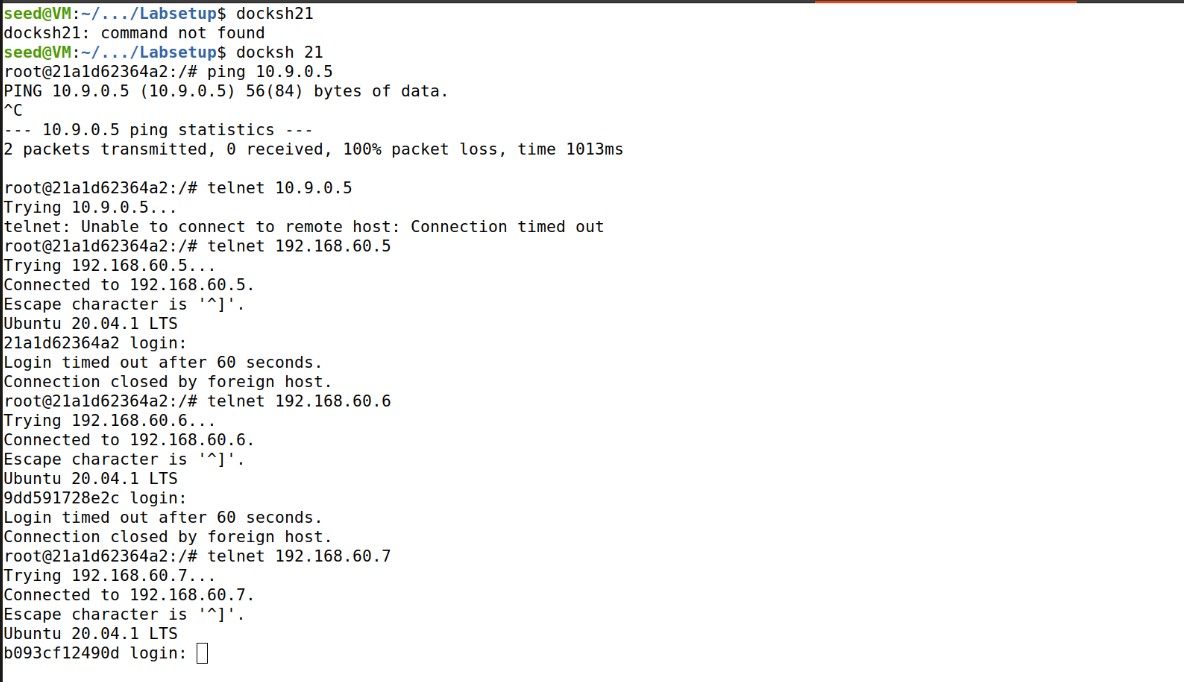
Checking telnet from internal host to external host :

Task 2.C: Protecting Internal Servers

All the internal hosts run a telnet server (listening to port 23). Outside hosts can only access the telnet server on 192.168.60.5, not the other internal hosts

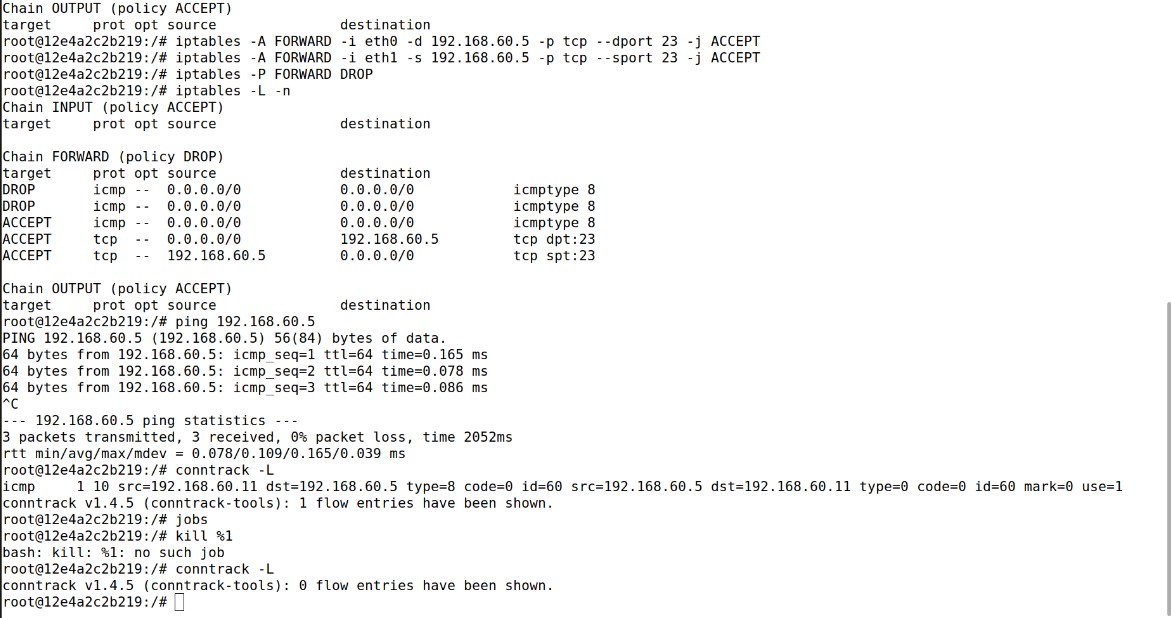
Host 10.9.0.5 can connect to 192.168.60.5.

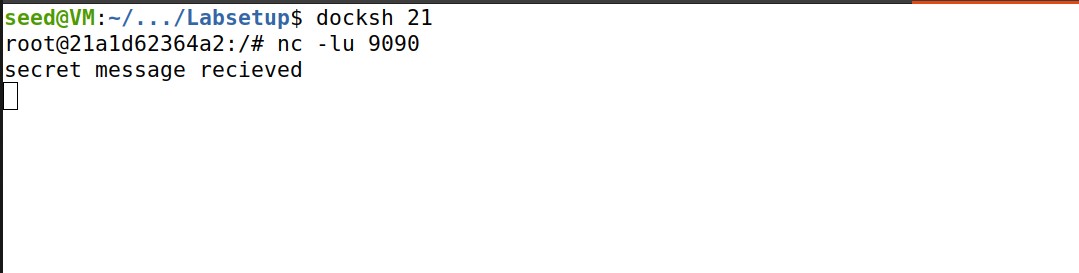
Outside hosts cannot access other internal servers

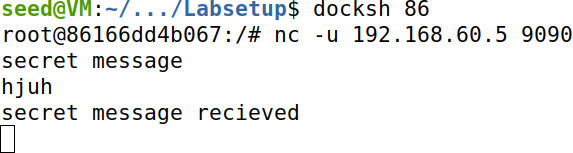
Internal hosts can access all the internal servers.

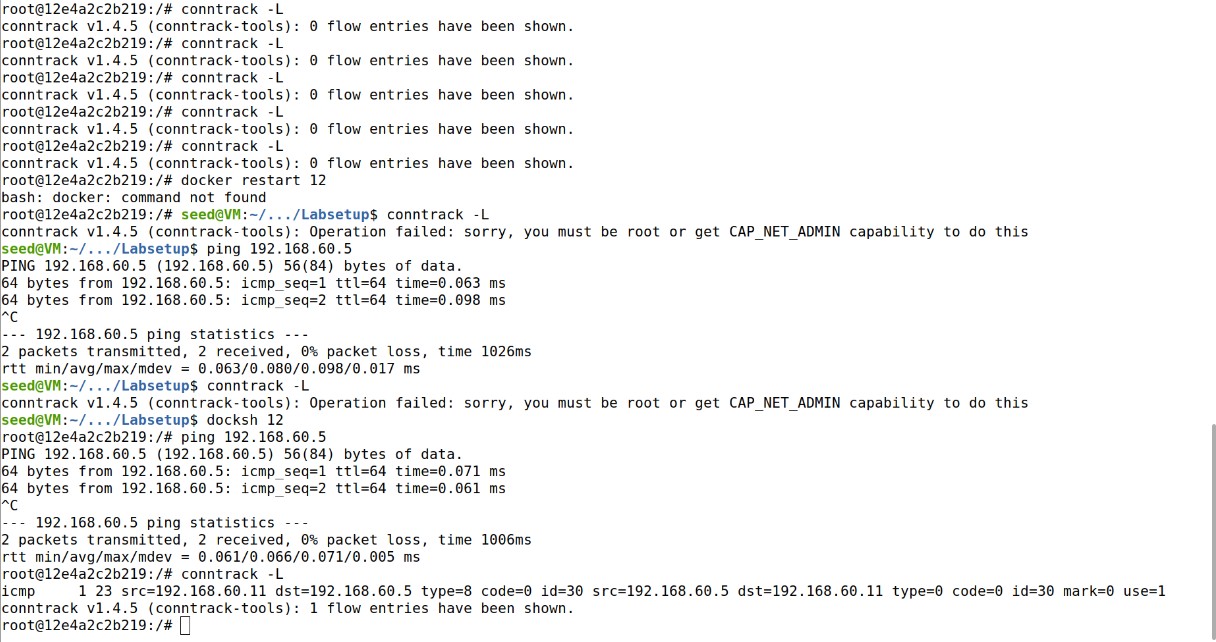
Internal hosts cannot access external servers.

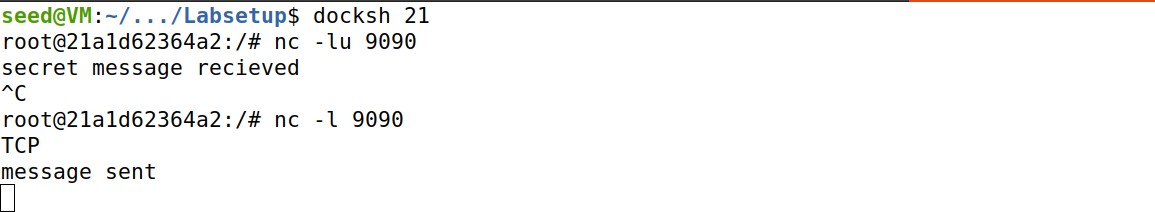
Task 3: Connection Tracking and Stateful Firewall

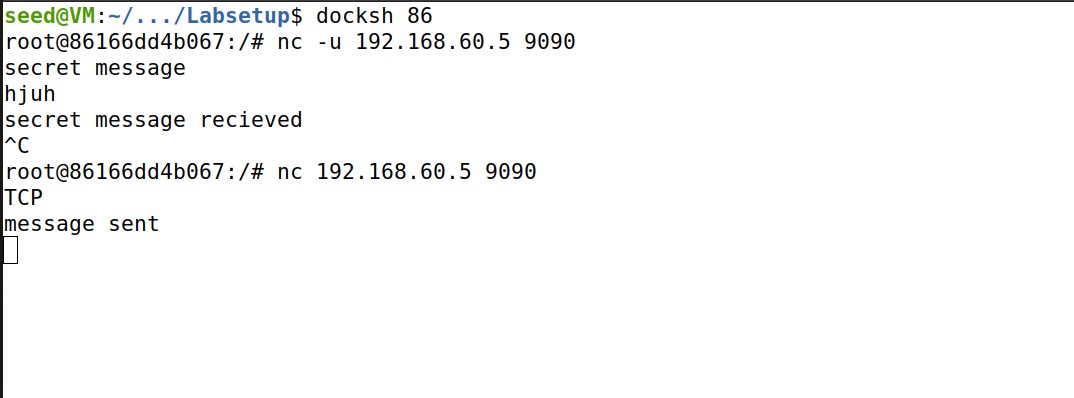
ICMP experiment: First ping 192.168.60.5 and check for time of connection using the command conntrack -L.

UDP experiment: on 192.168.60.5 run the command nc -lu 9090

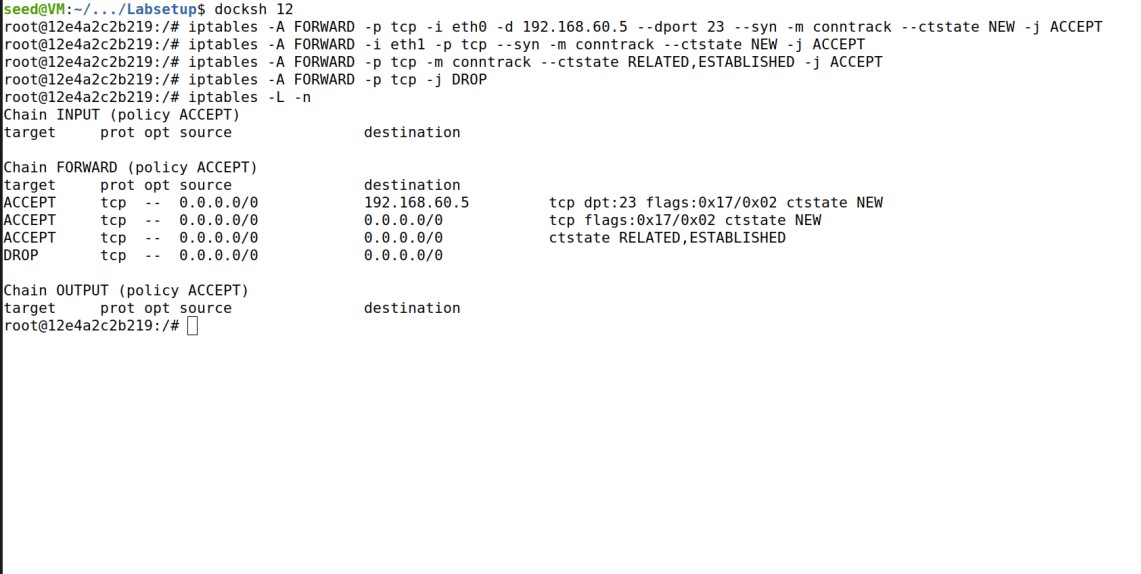
On 10.9.0.5 sending the UDP packets (hello) using nc -u 192.168.60.5 9090

Checking the connection request time using conntrack -L

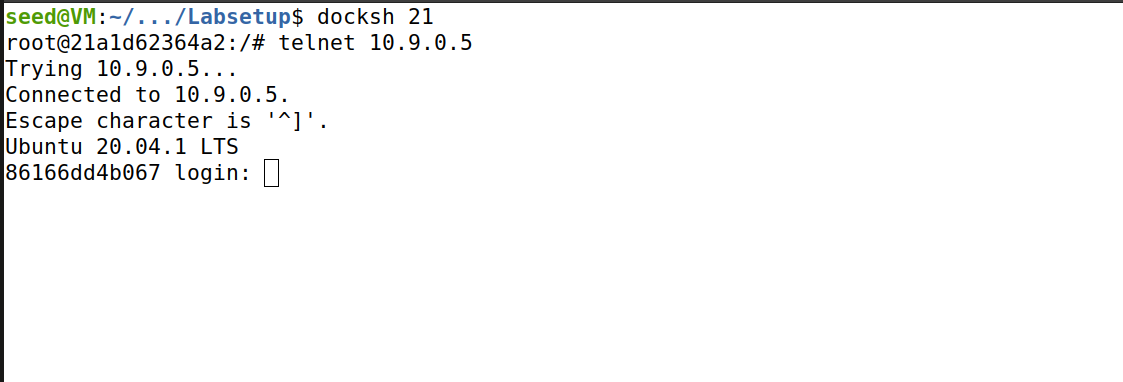
TCP experiment : On 192.168.60.5 starting TCP connection using nc -l 9090 ( hey message received after next step)

On 10.9.0.5 sending TCP packet using nc 192.168.60.5 9090 and

sending message.

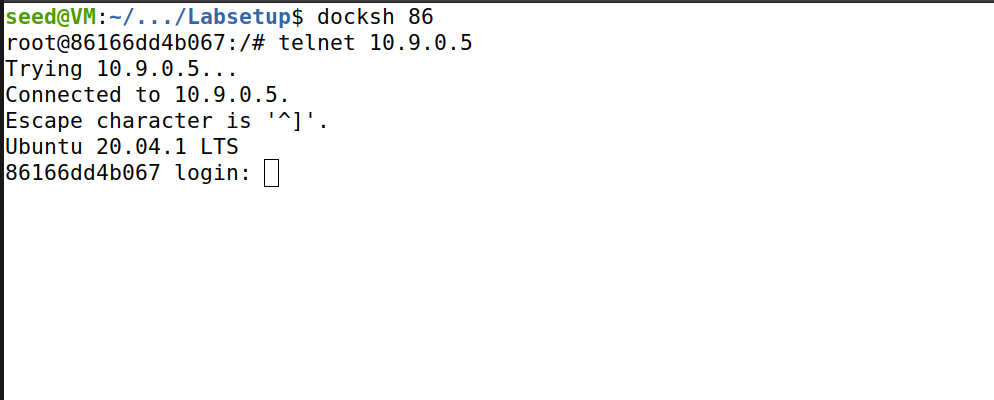
Task 3.B: Setting Up a Stateful Firewall

Setting iptables rules for allowing TCP packets

Telnet 10.9.0.5 from

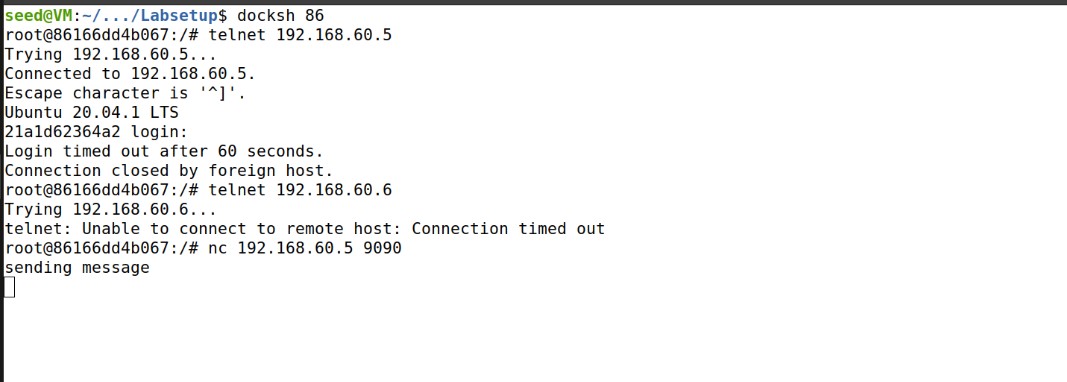
host 1 (192.168.60.5 )

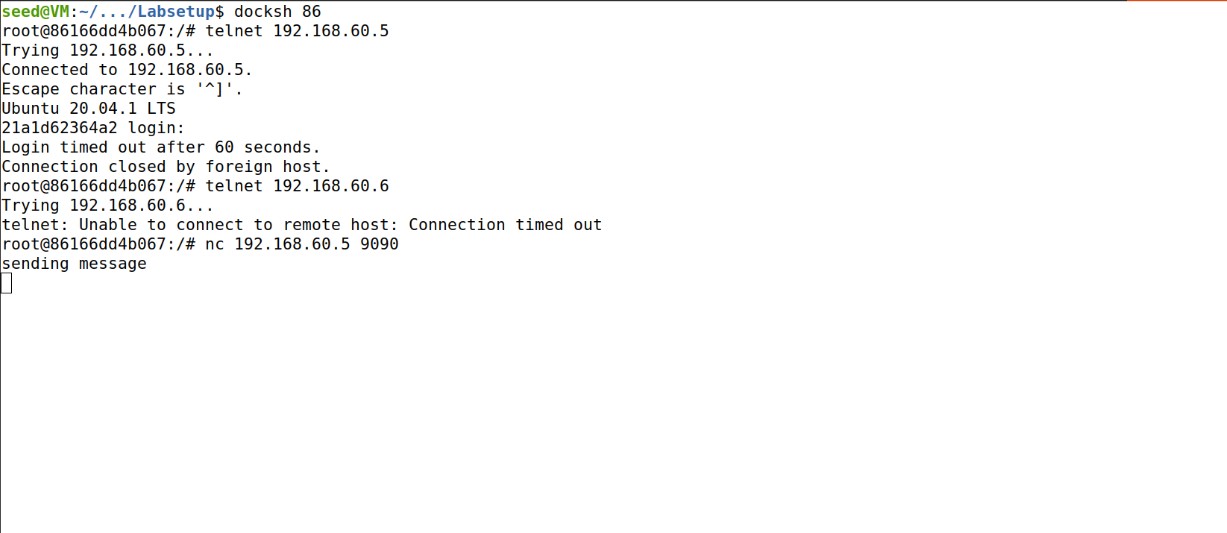
connection is established.

Telnet 192.168.60.5

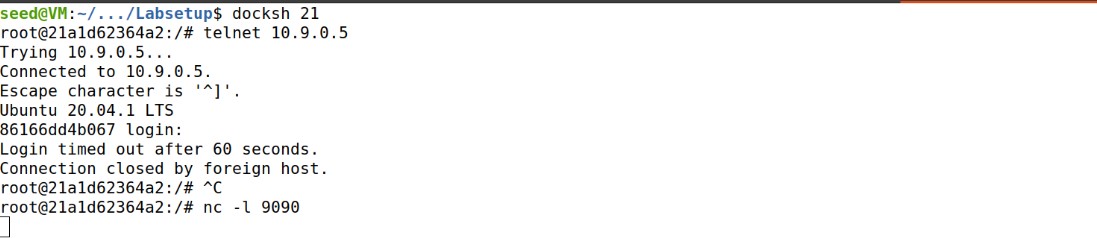
from hostA (10.9.0.5)

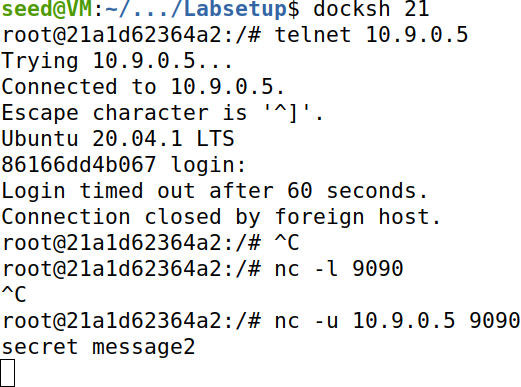
connection establishes

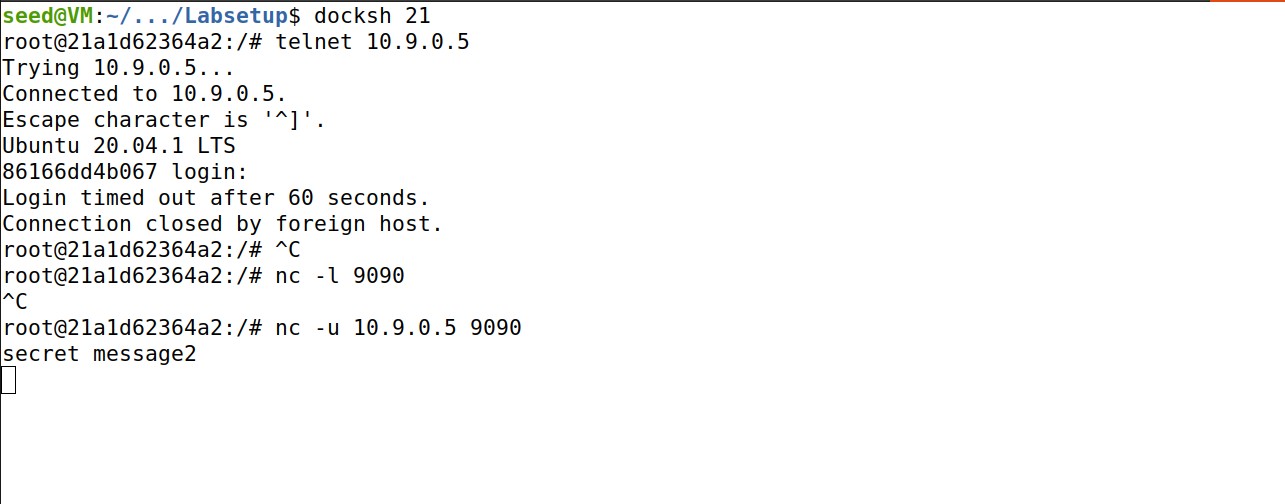
Telnet 192.168.60.6 from hostA connection fails due to iptables rules.

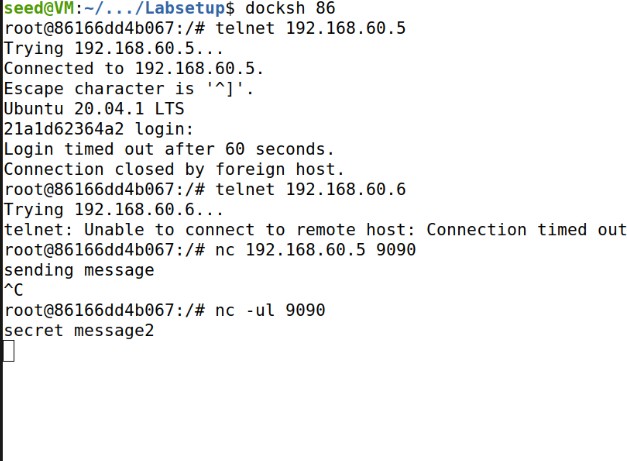
Run commands nc -l 9090 on host 1 and nc 192.168.60.5 9090 on host A,

with message to check connection.

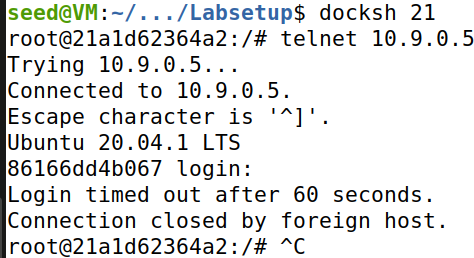
But nothing is received over TCP connection.

Run n -l 9090 on hostA, and run nc -u 10.9.0.5 9090 with message hello.

This receives message as packets are sent over UDP.

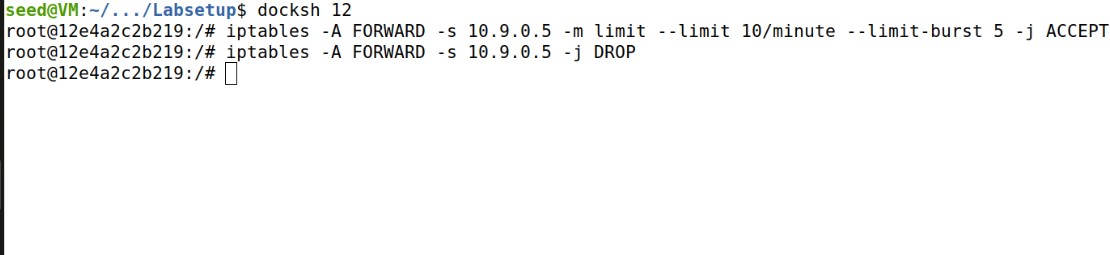
Telnet 192.168.60.5 from host A -connection works.

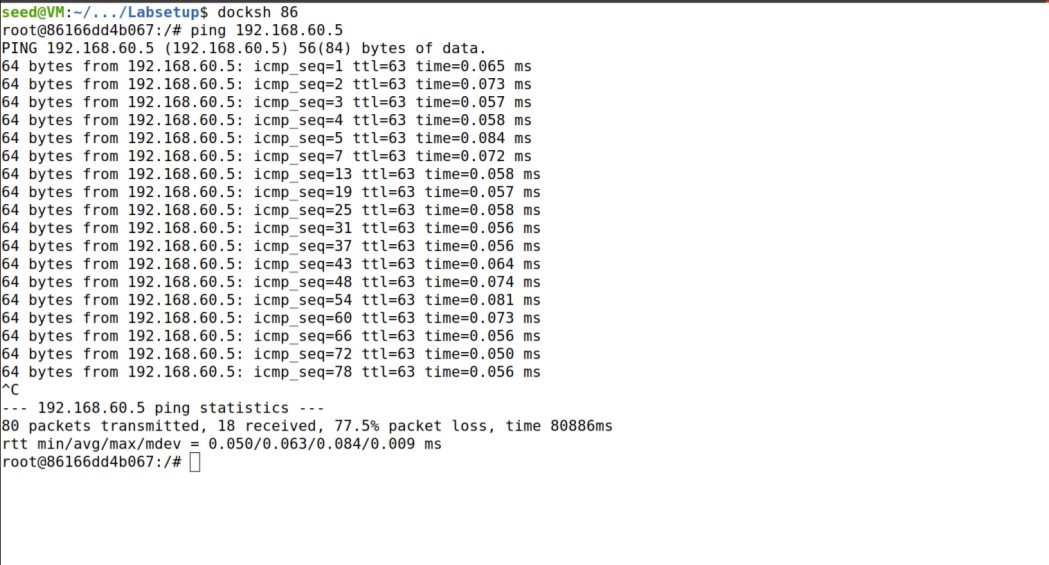
Telnet 192.168.60.6 from host A -connection will not work.

Telnet 10.9.0.5 from host1 –

connection works.

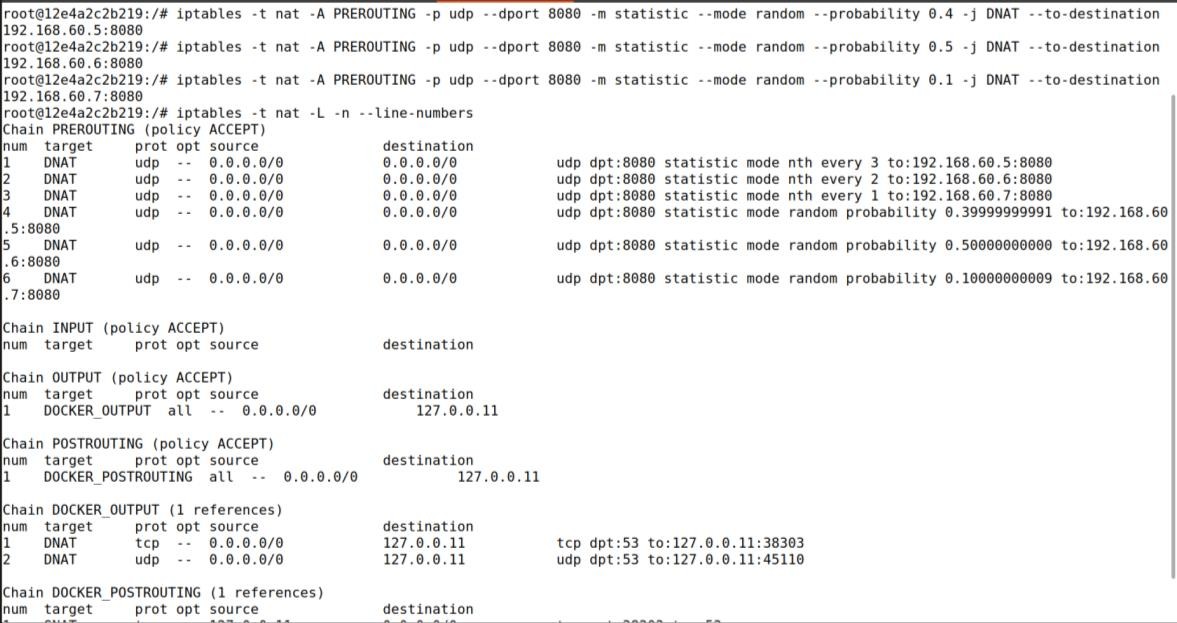
Task 4: Limiting Network Traffic

Running the below command on router to limit the number packet flow from external to internal host

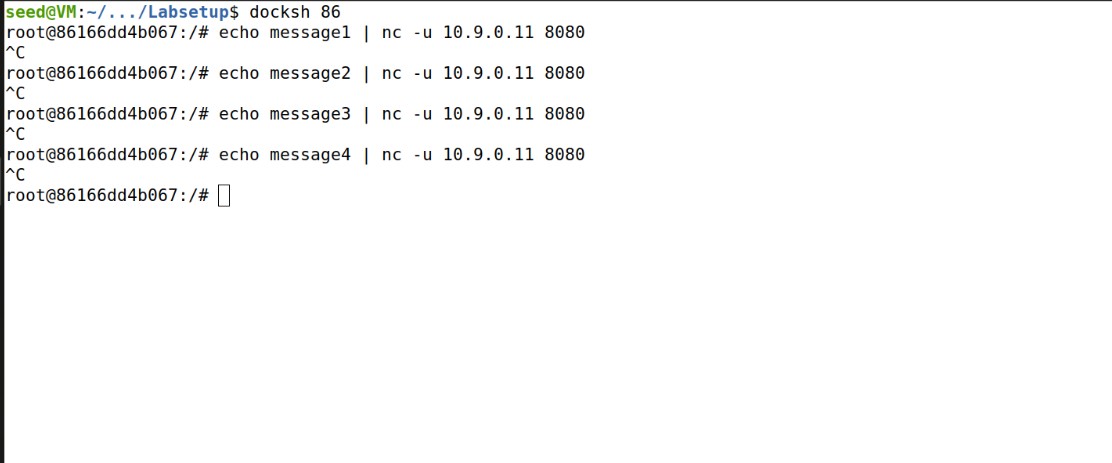
Login into external server

10.9.0.5 using docksh command and ping 1920.68.60.5 to check packet limit.

Task 5: Load Balancing

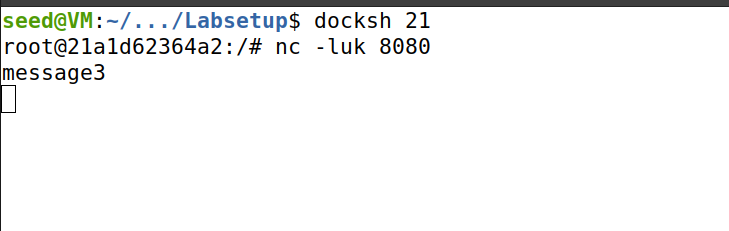
Using the nth mode (round-robin)

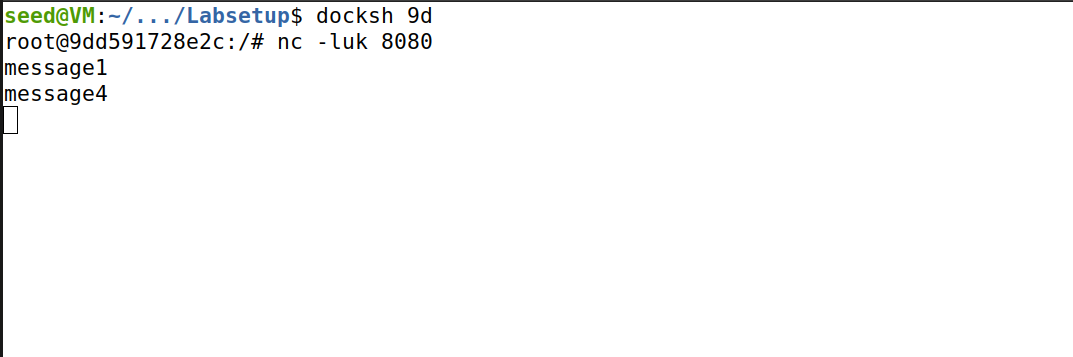
Using statistics module modifying packet journey to specific host using following commands on router.

Now using below command on host

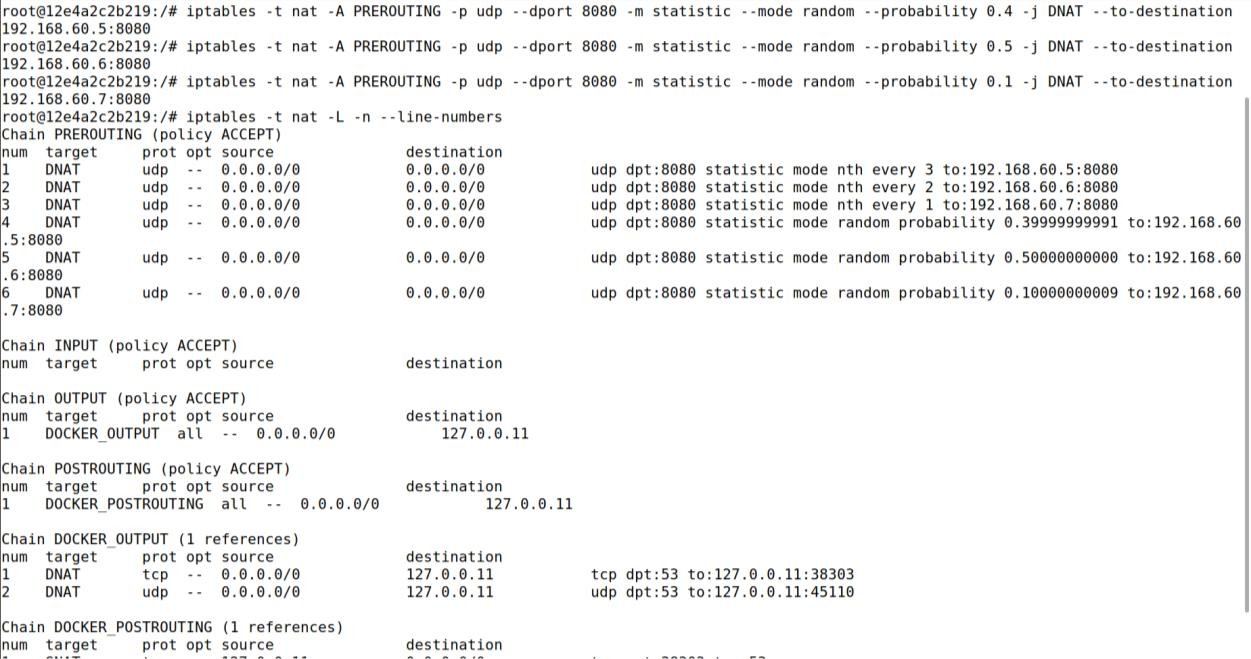
10.9.0.5 sending packets to internal server according to iptables rule

## echo hello | nc -u 10.9.0.11 8080

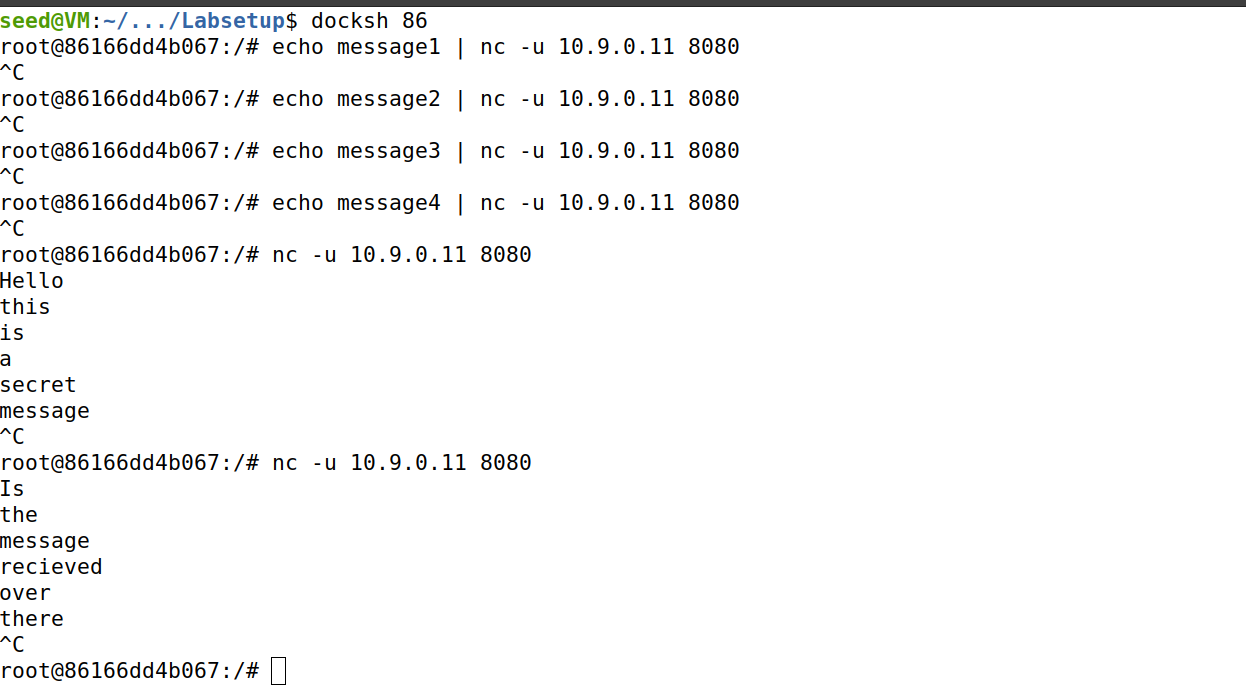
Every 3rd packet is sent to 192.168.60.5

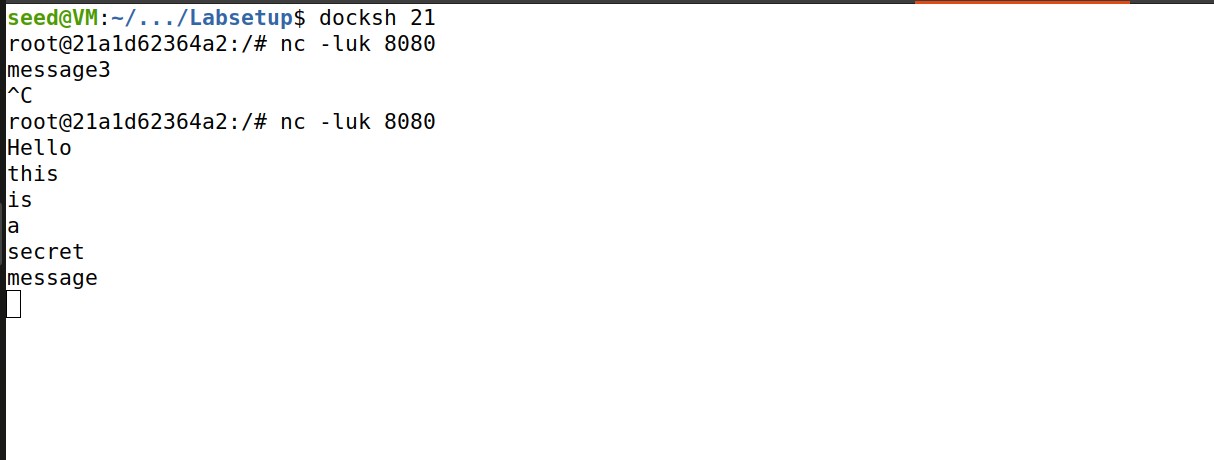
Every 2nd packet is sent to 192.168.60.6

Every 1st packet is sent to 192.168.60.7

Using the random mode

Select a matching packet with the probability mentioned as below command

Sending echo from host 10.9.0.11

Checking output on internal servers:

