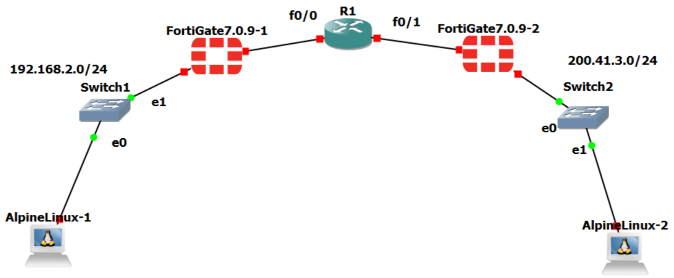
Part B Network IP spoofing Report

Group 4

Chennamadhava Sri Vibhav Raju	**11754947**
Sai Vijay Sankar Bheemana	**11846526**
Sowgoto Raha Sunny	**11762337**

Network Topology



connected by a Cisco router

Left Side – Network A

Network: 192.168.2.0/24

Switch1 connects to:

AlpineLinux-1 (attacker machine)

Router interface f0/0

Right Side – Network B

Network: 200.41.3.0/24

Switch2 connects to:

AlpineLinux-2 (victim machine)

Router interface f0/1

So the final layout exactly is what shown in screenshot

AlpineLinux-1 ---- Switch1-- firewall ---- R1 --- firewall -- Switch2 --- AlpineLinux-2

192.168.2.0/24 f0/0 R1 f0/1 200.41.3.0/24

We have **IDS** as **WireShark** and **IPS** as **ACL Access-Control Lists** in Router in our network

Main goal of the Lab is to demonstrates **TWO** things:

A. How to perform IP spoofing using Linux iptables this is done by modifying outgoing packets to fake the source IP

B. How to prevent spoofed packets using Cisco Extended ACL done by block any packet with a fake spoofed source IP

For this we have installed iptables on Alpine Linux both attacker and victim

Which they do NOT include iptables by default.

Commands **: apk add iptables**

Verified installation
iptables -L

We Assign the Correct IP or Static IP Addresses

AlpineLinux-1 (Attacker)

For this we edited this file

/etc/network/interfaces

Then set static IP:

address 192.168.2.2

netmask 255.255.255.0

gateway 192.168.2.1

AlpineLinux-2 (Victim)

Similarly:

address 200.41.3.2

netmask 255.255.255.0

gateway 200.41.3.1

While configuring Router R1 IP Configuration

We set interface f0/0

ip address 192.168.2.1 255.255.255.0

interface f0/1

ip address 200.41.3.1 255.255.255.0

For proper network routing

We have Tested this using Normal Connection

From AlpineLinux-1 → AlpineLinux-2

ping 200.41.3.2 works good

Router forwards packets properly, ICMP packets flowing normally through our created network

Source: 192.168.2.2

Destination: 200.41.3.2

Performing IP SPOOFING

This is the MOST IMPORTANT part in our attack simulation

We use iptables NAT POSTROUTING SNAT to change the source IP of outgoing packets from machine 1 alphine

command he used

iptables -t nat -A POSTROUTING -p icmp -j SNAT --to-source 246.79.20

we use some spoof IP here

Meaning of the command

-t nat → use NAT table

POSTROUTING → modify packets just before they leave

-p icmp → apply only to ping packets

SNAT → change source IP

--to-source → new spoof source IP

When we do this above steps and

```
ping 200.41.3.2
```

Now on Wireshark router or victim side the ICMP packet shows:

Source IP = 246.79.20 (spoof addr)

Destination = 200.41.3.2 (victim IP)

Proving spoofing is successful in our case

Meaning -> AlpineLinux-1 is pretending to be some other machine.

Router and victim have no idea it is a fake or spoofed address

Prevents IP Spoofing

We configure an Extended Access Control List on R1 router

Create ACL 110

```
access-list 110 permit ip 192.168.2.0 0.0.0.255 any
access-list 110 deny   ip any any
```

this only allow packets with legitimate source IP from the left LAN

Deny ALL other sources including spoofed ones

We have applied ACL to Router Interface f0/1 (outbound)

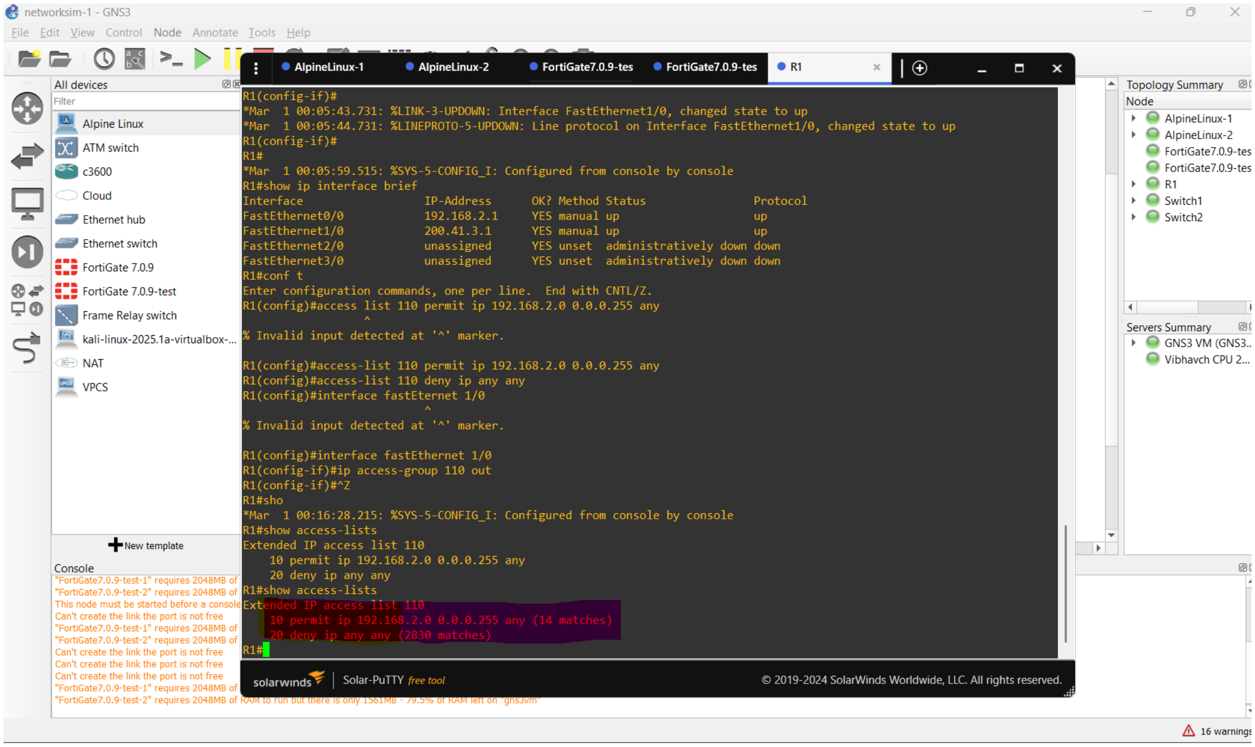
```
interface f0/1
```

While testing Spoofing Again its successfully blocked

spoofing ping from AlpineLinux-1, Router drops spoofed packets

Wireshark shows NO spoofed packets reaching victim while in capture

ACL hit-count REVEALS incrementation suggesting blockage



```
R1(config-if)#
*Mar  1 00:05:43.731: %LINK-3-UPDOWN: Interface FastEthernet1/0, changed state to up
*Mar  1 00:05:44.731: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to up
R1(config-if)#
R1#
*Mar  1 00:05:59.515: %SYS-5-CONFIG_I: Configured from console by console
R1#show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            192.168.2.1     YES manual up                    up
FastEthernet1/0            200.41.3.1      YES manual up                    up
FastEthernet2/0            unassigned      YES unset  administratively down down
FastEthernet3/0            unassigned      YES unset  administratively down down
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access list 110 permit ip 192.168.2.0 0.0.0.255 any
                  ^
% Invalid input detected at '^' marker.

R1(config)#access-list 110 permit ip 192.168.2.0 0.0.0.255 any
R1(config)#access-list 110 deny ip any any
R1(config)#interface fastEternet 1/0
                           ^
% Invalid input detected at '^' marker.

R1(config)#interface fastEthernet 1/0
R1(config-if)#ip access-group 110 out
R1(config-if)#^Z
R1#sho
*Mar  1 00:16:28.215: %SYS-5-CONFIG_I: Configured from console by console
R1#show access-lists
Extended IP access list 110
    10 permit ip 192.168.2.0 0.0.0.255 any
    20 deny ip any any
R1#show access-lists
Extended IP access list 110
    10 permit ip 192.168.2.0 0.0.0.255 any (14 matches)
    20 deny ip any any (2830 matches)
R1#
```

```
"FortiGate7.0.9-test-1" requires 2048MB of
"FortiGate7.0.9-test-2" requires 2048MB of
This node must be started before a console
Can't create the link the port is not free
"FortiGate7.0.9-test-1" requires 2048MB of
"FortiGate7.0.9-test-2" requires 2048MB of
Can't create the link the port is not free
Can't create the link the port is not free
Can't create the link the port is not free
"FortiGate7.0.9-test-1" requires 2048MB of
"FortiGate7.0.9-test-2" requires 2048MB of RAM to run but there is only 1561MB - 79.5% of RAM left on "gns3vm"
```

show access-lists 110

"deny 2830 matches..."

Confirming the ACL is blocking spoofed packets.

If in case we removing the Spoof Rule

iptables -t nat -D POSTROUTING -p icmp -j SNAT --to-source 246.79.20

Then ping works normally.

ACL no longer blocks because packets now come with correct source IP instead of spoofed.