Student Name: Shachi Shah

Email: spshah22@asu.edu

Submission Date: 05/22/2025

Class Name and Term: CSE548 Summer 2025

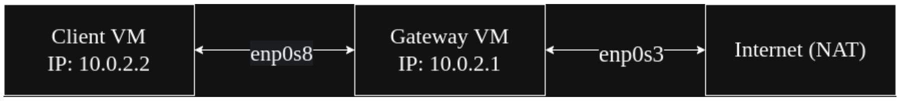
Packet Filter Firewall (iptables) Project

I. Project Overview

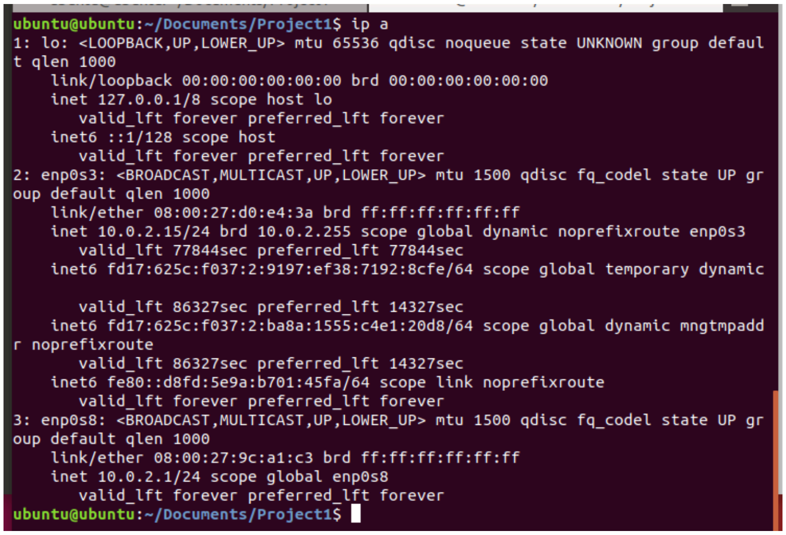
This project involved configuring a packet filter firewall in a two-VM network setup using iptables. The Gateway VM acted as a NAT gateway and HTTP server, while the Client VM accessed external and internal resources through it. The main goal was to implement a secure firewall with default-deny rules, only permitting necessary traffic like DNS, HTTP, and pings to 8.8.8.8 while blocking others. Screenshots were taken throughout to verify functionality. The final configuration ensured that only essential traffic was allowed, complying with all evaluation criteria.

II. Network Setup

- Network Diagram: This network topology was designed to isolate the internal communication between the client and the gateway on a private interface (enp0s8), while allowing the gateway to reach the internet via NAT (enp0s3). The client sends all its internet-bound traffic through the gateway, which performs IP masquerading. This setup was essential for testing the firewall rules in a controlled environment, ensuring proper traffic filtering and NAT behavior.

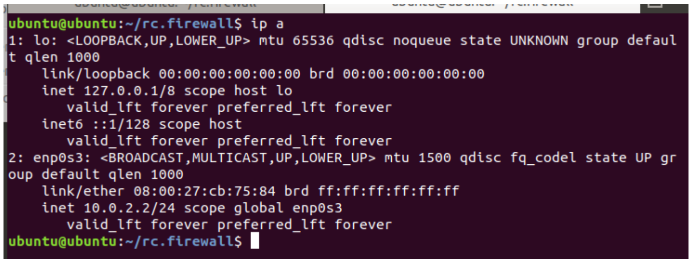


- Topology: Two Ubuntu VMs in VirtualBox
    - Gateway VM: Dual interfaces
    - enp0s3: NAT (for external access)
    - enp0s8: Internal network (IP: 10.0.2.1)
- Client VM: Single Interface (enp0s3, IP: 10.0.2.2)
- Initial Reachability:
    - Client could reach Gateway and 8.8.8.8 before the firewall hardened.
- Relevant Screenshots:
    - Gateway ip a



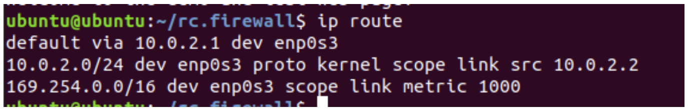
```
ubuntu@ubuntu:~/Documents/Project1$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:d0:e4:3a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 77844sec preferred_lft 77844sec
    inet6 fd17:625c:f037:2:9197:ef38:7192:8cfe/64 scope global temporary dynamic

       valid_lft 86327sec preferred_lft 14327sec
    inet6 fd17:625c:f037:2:ba8a:1555:c4e1:20d8/64 scope global dynamic mngtmpadd
r noprefixroute
       valid_lft 86327sec preferred_lft 14327sec
    inet6 fe80::d8fd:5e9a:b701:45fa/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:9c:a1:c3 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.1/24 scope global enp0s8
       valid_lft forever preferred_lft forever
ubuntu@ubuntu:~/Documents/Project1$
```

- Client ip a



```
ubuntu@ubuntu:~/rc.firewall$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:cb:75:84 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.2/24 scope global enp0s3
       valid_lft forever preferred_lft forever
ubuntu@ubuntu:~/rc.firewall$
```

- ○ Client ip route



```
ubuntu@ubuntu:~/rc.firewall$ ip route
default via 10.0.2.1 dev enp0s3
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.2
169.254.0.0/16 dev enp0s3 scope link metric 1000
```

III. Software

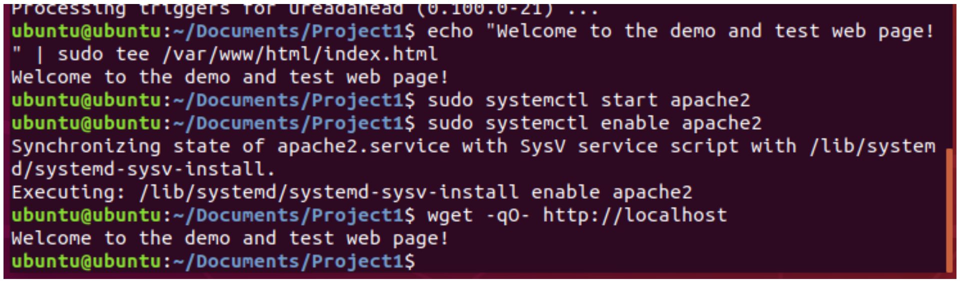
This project utilized several core tools and services on Ubuntu Linux:

- iptables - to implement packet filtering rules on the Gateway VM.
- Apache2 - to host a demo web page on the Gateway VM.
- wget - to fetch web content from both localhost and across the VMs.
- ping - to test ICMP echo reply access between nodes and to the public internet.
- nmap - to scan TCP/UDP ports and verify service access restrictions.
- arping and iproute2 - used for low-level diagnostics and confirming network interface behavior.

IV. Project Description

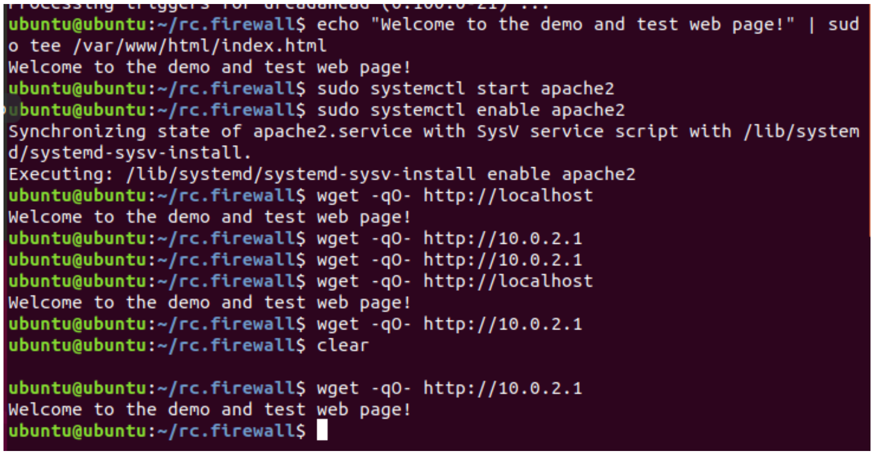
The following steps outline how the packet filter firewall and NAT configuration were implemented:

- Apache Setup on Gateway
  - Installed and enabled Apache2 via `sudo apt install apache2`
  - Edited `/var/www/html/index.html` to display a custom welcome message
  - Verified using:
    - `wget -qO- http://localhos`t (Screenshot: Gateway Apache Test.png)



```
Processing triggers for ureadahead (0.100.0-21) ...
ubuntu@ubuntu:~/Documents/Project1$ echo "Welcome to the demo and test web page!
" | sudo tee /var/www/html/index.html
Welcome to the demo and test web page!
ubuntu@ubuntu:~/Documents/Project1$ sudo systemctl start apache2
ubuntu@ubuntu:~/Documents/Project1$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/system
d/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
ubuntu@ubuntu:~/Documents/Project1$ wget -qO- http://localhost
Welcome to the demo and test web page!
ubuntu@ubuntu:~/Documents/Project1$
```

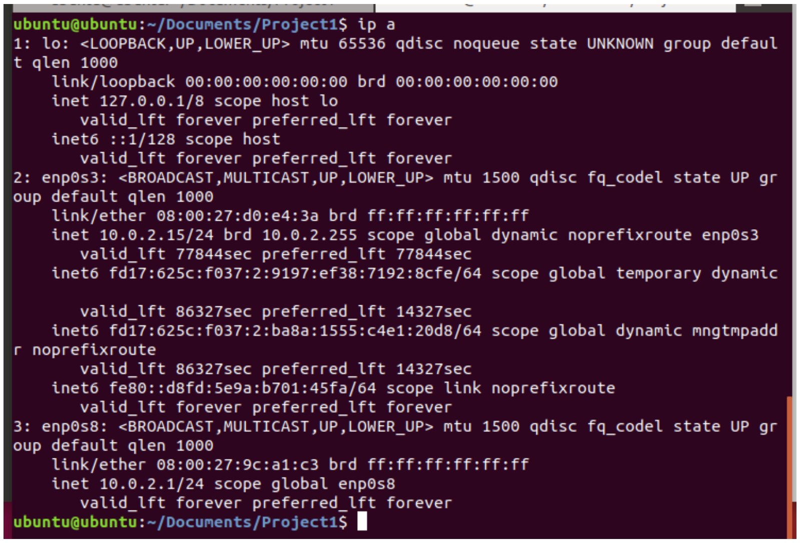
    - `wget -qO- http://10.0.2.1` from client (Screenshot: Client Apache Test.png)



```
Processing triggers for ureadahead (0.100.0-21) ...
ubuntu@ubuntu:~/rc.firewall$ echo "Welcome to the demo and test web page!" | sud
o tee /var/www/html/index.html
Welcome to the demo and test web page!
ubuntu@ubuntu:~/rc.firewall$ sudo systemctl start apache2
ubuntu@ubuntu:~/rc.firewall$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/system
d/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://localhost
Welcome to the demo and test web page!
ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://10.0.2.1
ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://10.0.2.1
ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://localhost
Welcome to the demo and test web page!
ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://10.0.2.1
ubuntu@ubuntu:~/rc.firewall$ clear

ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://10.0.2.1
Welcome to the demo and test web page!
ubuntu@ubuntu:~/rc.firewall$
```

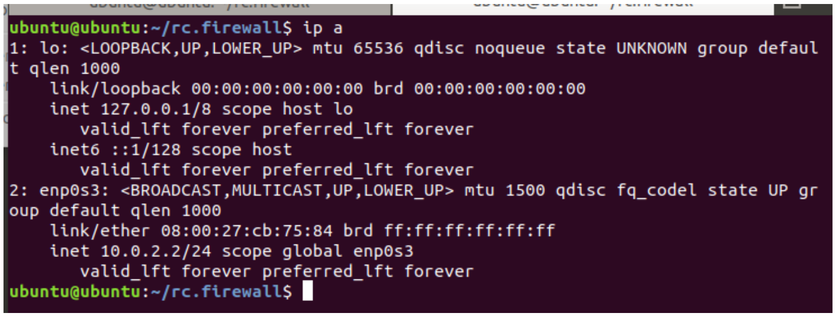
- Interface and Routing Configuration
  - Manually configured static IPs:

- Gateway internal IP: 10.0.2.1 (Screenshot: Gateway ip a.png)



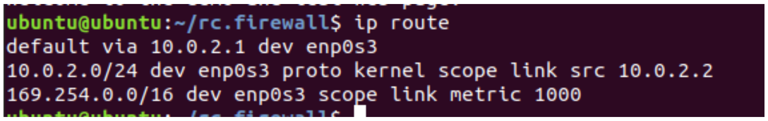
```
ubuntu@ubuntu:~/Documents/Project1$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:d0:e4:3a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 77844sec preferred_lft 77844sec
    inet6 fd17:625c:f037:2:9197:ef38:7192:8cfe/64 scope global temporary dynamic

       valid_lft 86327sec preferred_lft 14327sec
    inet6 fd17:625c:f037:2:ba8a:1555:c4e1:20d8/64 scope global dynamic mngtmpadd
r noprefixroute
       valid_lft 86327sec preferred_lft 14327sec
    inet6 fe80::d8fd:5e9a:b701:45fa/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:9c:a1:c3 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.1/24 scope global enp0s8
       valid_lft forever preferred_lft forever
ubuntu@ubuntu:~/Documents/Project1$
```

- Client IP: 10.0.2.2 (Screenshot: Client ip a.png)



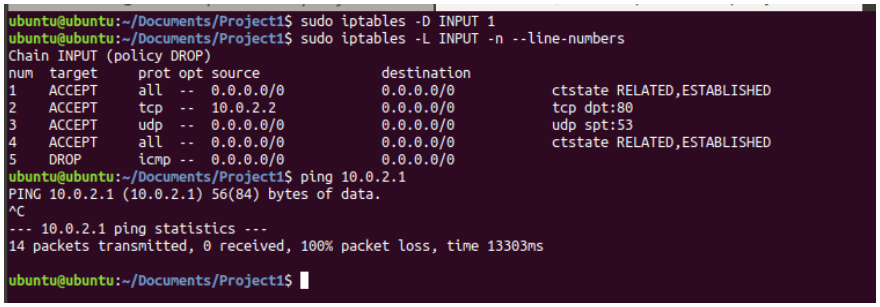
```
ubuntu@ubuntu:~/rc.firewall$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:cb:75:84 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.2/24 scope global enp0s3
       valid_lft forever preferred_lft forever
ubuntu@ubuntu:~/rc.firewall$
```

- Verified routing via ip route (Screenshot: Client ip route.png)



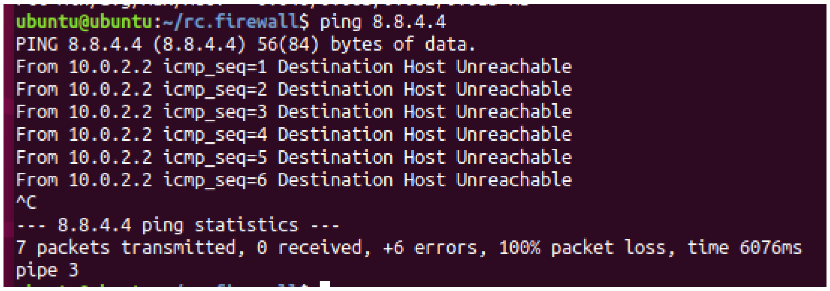
```
ubuntu@ubuntu:~/rc.firewall$ ip route
default via 10.0.2.1 dev enp0s3
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.2
169.254.0.0/16 dev enp0s3 scope link metric 1000
```

- Firewall Configuration
  - Used iptables to:
    - Set default policies to DROP for INPUT, OUTPUT, FORWARD
    - Allow only:
      - Loopback traffic (localhost)
      - Outgoing HTTP/HTTPS, DNS, and ping to 8.8.8.8 from client
      - Incoming HTTP from client (port 80)
      - Masquerading for NAT using `iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE`
    - Block:
      - All pings to gateway (Screenshot: Client ping 10.0.2.1.png)



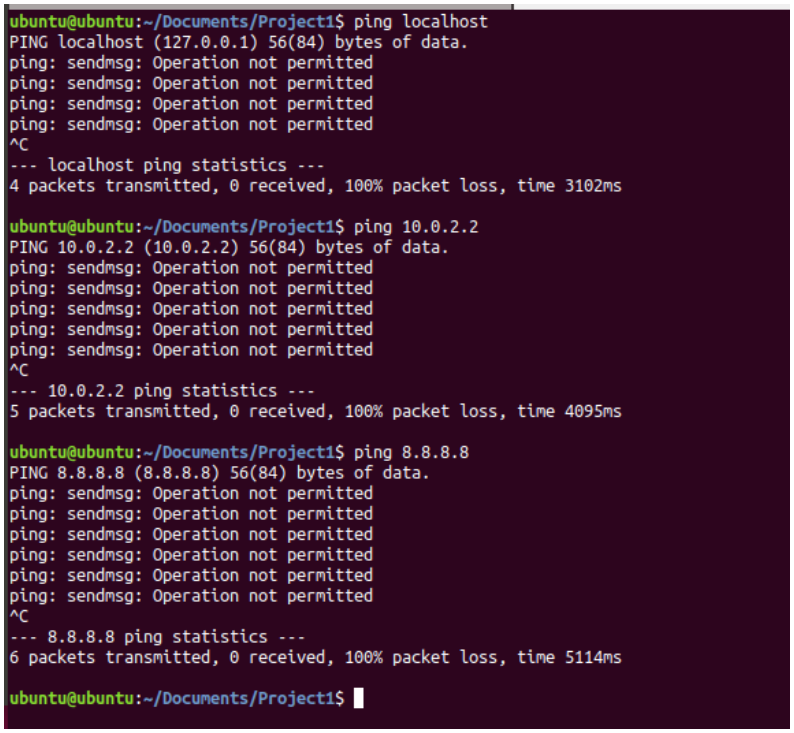
```
ubuntu@ubuntu:~/Documents/Project1$ sudo iptables -D INPUT 1
ubuntu@ubuntu:~/Documents/Project1$ sudo iptables -L INPUT -n --line-numbers
Chain INPUT (policy DROP)
num  target     prot opt source               destination
1    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABLISHED
2    ACCEPT     tcp  --  10.0.2.2             0.0.0.0/0            tcp dpt:80
3    ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0            udp spt:53
4    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABLISHED
5    DROP       icmp --  0.0.0.0/0            0.0.0.0/0
ubuntu@ubuntu:~/Documents/Project1$ ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1) 56(84) bytes of data.
^C
--- 10.0.2.1 ping statistics ---
14 packets transmitted, 0 received, 100% packet loss, time 13303ms

ubuntu@ubuntu:~/Documents/Project1$
```

      - Ping to other public DNS like 8.8.4.4 (Screenshot: Client ping 8.8.4.4.png)



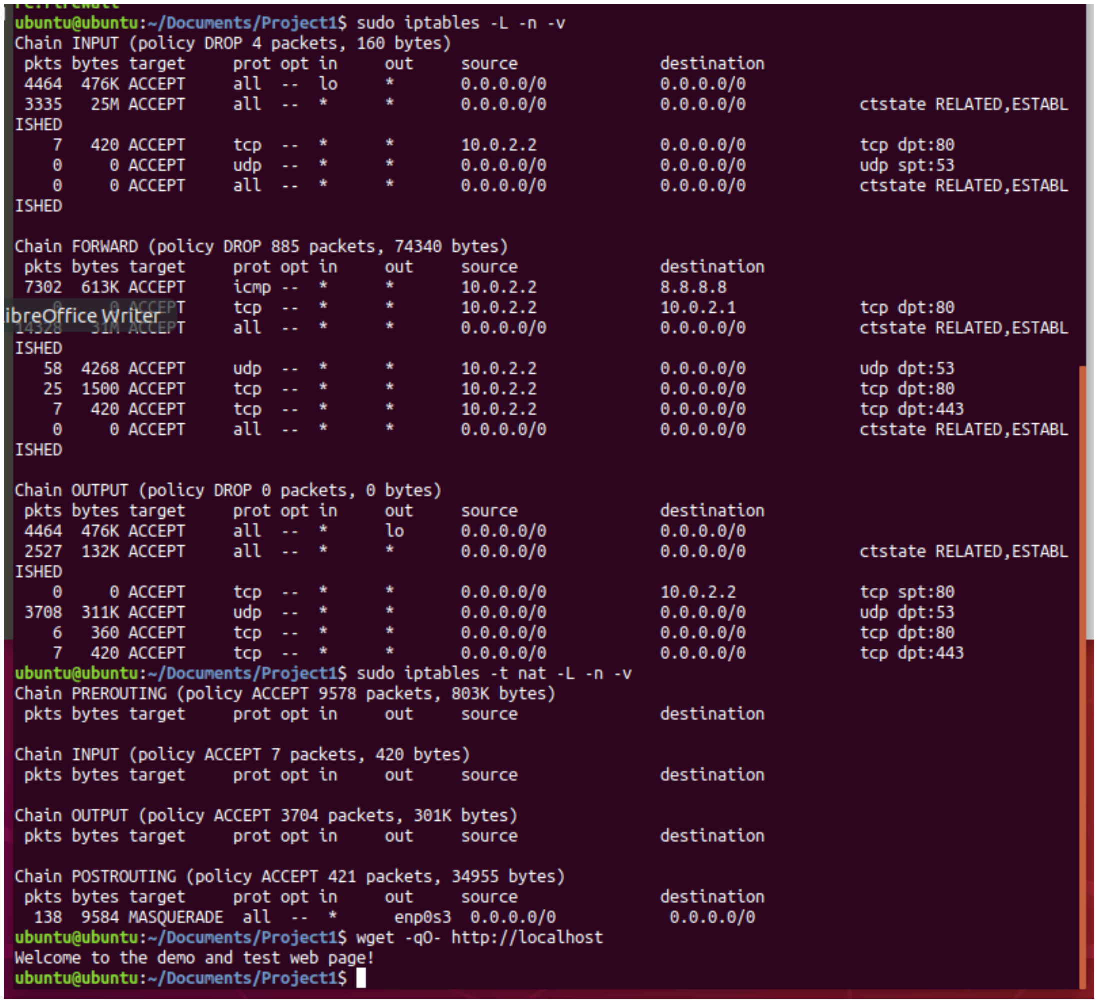
```
ubuntu@ubuntu:~/rc.firewall$ ping 8.8.4.4
PING 8.8.4.4 (8.8.4.4) 56(84) bytes of data.
From 10.0.2.2 icmp_seq=1 Destination Host Unreachable
From 10.0.2.2 icmp_seq=2 Destination Host Unreachable
From 10.0.2.2 icmp_seq=3 Destination Host Unreachable
From 10.0.2.2 icmp_seq=4 Destination Host Unreachable
From 10.0.2.2 icmp_seq=5 Destination Host Unreachable
From 10.0.2.2 icmp_seq=6 Destination Host Unreachable
^C
--- 8.8.4.4 ping statistics ---
7 packets transmitted, 0 received, +6 errors, 100% packet loss, time 6076ms
pipe 3
```

- All pings from gateway (Screenshot: Gateway All Pings.png)



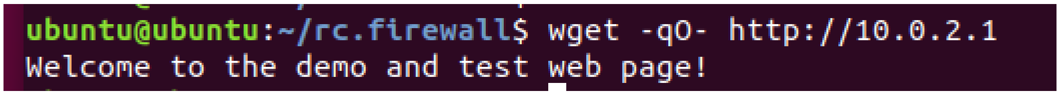
```
ubuntu@ubuntu:~/Documents/Project1$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- localhost ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3102ms

ubuntu@ubuntu:~/Documents/Project1$ ping 10.0.2.2
PING 10.0.2.2 (10.0.2.2) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 10.0.2.2 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4095ms

ubuntu@ubuntu:~/Documents/Project1$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5114ms

ubuntu@ubuntu:~/Documents/Project1$
```

- Full ruleset saved via iptables-save (Screenshot: Gateway iptables.png)



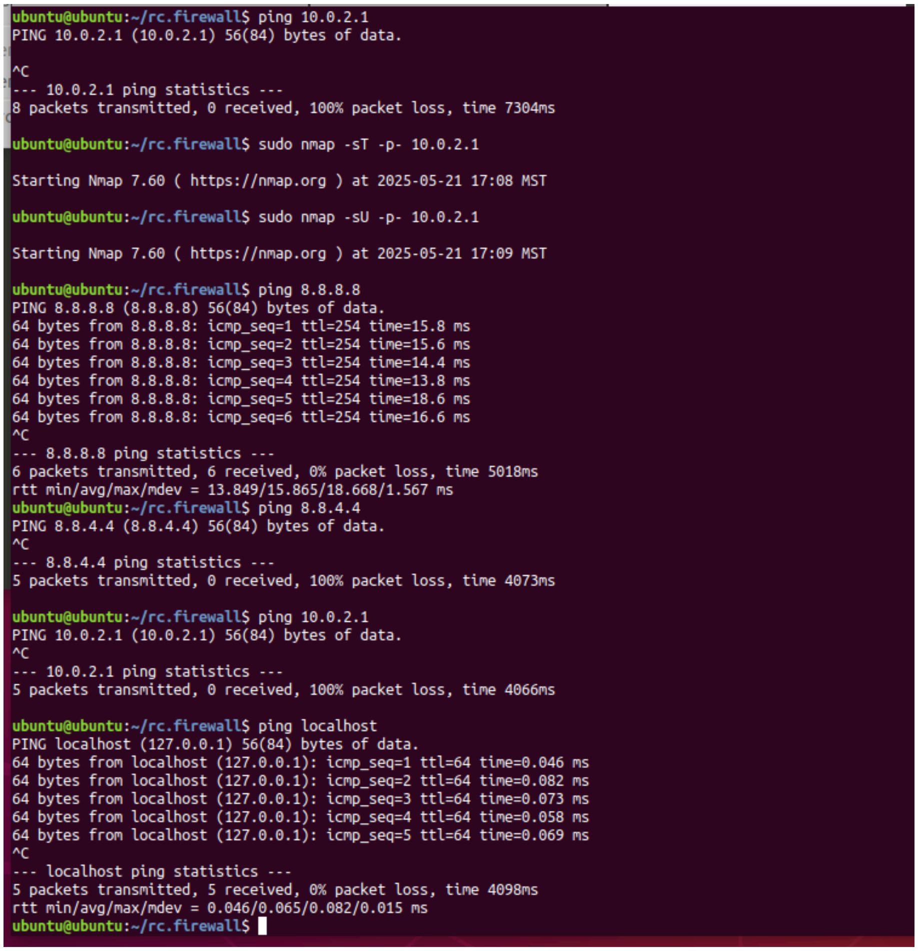
```
ubuntu@ubuntu:~/Documents/Project1$ sudo iptables -L -n -v
Chain INPUT (policy DROP 4 packets, 160 bytes)
 pkts bytes target     prot opt in     out     source               destination         
 4464  476K ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0           
 3335   25M ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABL
ISHED
    7   420 ACCEPT     tcp  --  *      *       10.0.2.2             0.0.0.0/0            tcp dpt:80
    0     0 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp spt:53
    0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABL
ISHED

Chain FORWARD (policy DROP 885 packets, 74340 bytes)
 pkts bytes target     prot opt in     out     source               destination         
 7302  613K ACCEPT     icmp --  *      *       10.0.2.2             8.8.8.8             
    0     0 ACCEPT     tcp  --  *      *       10.0.2.2             10.0.2.1             tcp dpt:80
14328   31M ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABL
ISHED
   58  4268 ACCEPT     udp  --  *      *       10.0.2.2             0.0.0.0/0            udp dpt:53
   25  1500 ACCEPT     tcp  --  *      *       10.0.2.2             0.0.0.0/0            tcp dpt:80
    7   420 ACCEPT     tcp  --  *      *       10.0.2.2             0.0.0.0/0            tcp dpt:443
    0     0 ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABL
ISHED

Chain OUTPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination         
 4464  476K ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0           
 2527  132K ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABL
ISHED
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            10.0.2.2             tcp spt:80
 3708  311K ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53
    6   360 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:80
    7   420 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:443
ubuntu@ubuntu:~/Documents/Project1$ sudo iptables -t nat -L -n -v
Chain PREROUTING (policy ACCEPT 9578 packets, 803K bytes)
 pkts bytes target     prot opt in     out     source               destination         

Chain INPUT (policy ACCEPT 7 packets, 420 bytes)
 pkts bytes target     prot opt in     out     source               destination         

Chain OUTPUT (policy ACCEPT 3704 packets, 301K bytes)
 pkts bytes target     prot opt in     out     source               destination         

Chain POSTROUTING (policy ACCEPT 421 packets, 34955 bytes)
 pkts bytes target     prot opt in     out     source               destination         
  138  9584 MASQUERADE  all  --  *      enp0s3  0.0.0.0/0            0.0.0.0/0           
ubuntu@ubuntu:~/Documents/Project1$ wget -qO- http://localhost
Welcome to the demo and test web page!
ubuntu@ubuntu:~/Documents/Project1$ 
```

- Client Testing
    - Verified HTTP access to gateway (Screenshot: Client wget -q0- http10.0.2.1.png)



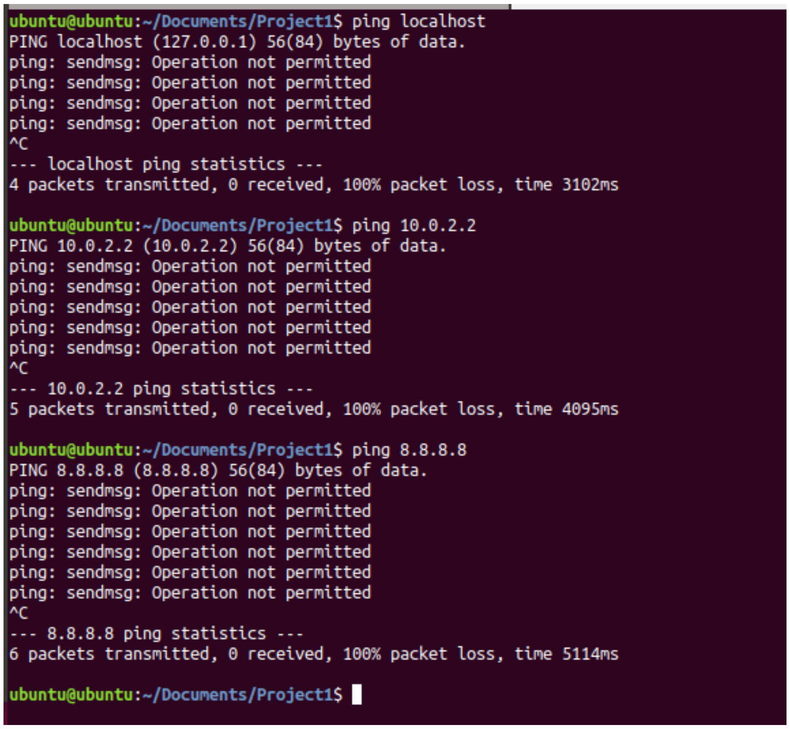
```
ubuntu@ubuntu:~/rc.firewall$ wget -qO- http://10.0.2.1
Welcome to the demo and test web page!
```

    - Confirmed access to 8.8.8.8 and failure to reach 8.8.4.4 and gateway (Screenshot: Client All Pings.png)



```
ubuntu@ubuntu:~/rc.firewall$ ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1) 56(84) bytes of data.

^C
--- 10.0.2.1 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7304ms

ubuntu@ubuntu:~/rc.firewall$ sudo nmap -sT -p- 10.0.2.1

Starting Nmap 7.60 ( https://nmap.org ) at 2025-05-21 17:08 MST

ubuntu@ubuntu:~/rc.firewall$ sudo nmap -sU -p- 10.0.2.1

Starting Nmap 7.60 ( https://nmap.org ) at 2025-05-21 17:09 MST

ubuntu@ubuntu:~/rc.firewall$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=254 time=15.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=254 time=15.6 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=254 time=14.4 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=254 time=13.8 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=254 time=18.6 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=254 time=16.6 ms
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5018ms
rtt min/avg/max/mdev = 13.849/15.865/18.668/1.567 ms
ubuntu@ubuntu:~/rc.firewall$ ping 8.8.4.4
PING 8.8.4.4 (8.8.4.4) 56(84) bytes of data.
^C
--- 8.8.4.4 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4073ms

ubuntu@ubuntu:~/rc.firewall$ ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1) 56(84) bytes of data.
^C
--- 10.0.2.1 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4066ms

ubuntu@ubuntu:~/rc.firewall$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.046 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.082 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.073 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.058 ms
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.069 ms
^C
--- localhost ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4098ms
rtt min/avg/max/mdev = 0.046/0.065/0.082/0.015 ms
ubuntu@ubuntu:~/rc.firewall$
```

  - Conducted nmap scans:
    - `sudo nmap -sT -p- 10.0.2.1` (only port 80 open)
    - `sudo nmap -sU -p- 10.0.2.1` (all ports closed)
- Gateway Testing
  - Verified ping to localhost, client, and 8.8.8.8 were blocked (Screenshot: Gateway All Pings.png)



```
ubuntu@ubuntu:~/Documents/Project1$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- localhost ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3102ms

ubuntu@ubuntu:~/Documents/Project1$ ping 10.0.2.2
PING 10.0.2.2 (10.0.2.2) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 10.0.2.2 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4095ms

ubuntu@ubuntu:~/Documents/Project1$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5114ms

ubuntu@ubuntu:~/Documents/Project1$
```

V. Conclusion

This project reinforced secure network design using iptables, ensuring only minimal required traffic is permitted. Key learnings included:

- Importance of iptables rule order and placement for correct traffic filtering.
- Differentiating traffic directions (INPUT, OUTPUT, FORWARD).
- Applying NAT/MASQUERADE to enable client's internet access via gateway.
- Iterative testing and logging to confirm compliance with all rubric rules.

Self-assessment: Fully implemented and tested the lab rubric requirements. All screenshots confirm firewall compliance and functionality per specification.

VI. Appendix B: Attached files

Screenshots:

- Client All Pings.png
- Client Apache Test.png
- Client ip a.png
- Client ip route.png
- Client ping 8.8.4.4.png
- Client ping 8.8.8.8.png
- Client ping 10.0.2.1.png
- Client ping google.com.png
- Client wget -q0- http10.0.2.1.png
- Gateway All Pings.png
- Gateway Apache Test.png
- Gateway ip a.png
- Gateway iptables.png

VII. References

[1] Nmap.org "Firewall/IDS Evasion and Spoofing." available at https://nmap.org/book/man-bypass-firewalls-ids.html, accessed by 08/21/2025

[2] Ubuntu, "Basic iptables HowTo" available at https://help.ubuntu.com/community/IptablesHowTo, accessed by 08/21/2025