# Securing Networks (Network based Firewall)

CaseStudy-2

**Taimur Shah**
**Mohamed Yasser Kaleelurrahman**
**Seneca College**

**5th December 2016**

# Contents

Objective ........ 3
Host Machine - 1 Specification ........ 3
Host Machine - 2 Specification ........ 3
Proof of Working Servers ........ 3
    Primary Server & Secondary Server (DNS) ........ 3
    Mail Server ........ 4
    Working Web Server ........ 4
VM Specifications & Access Policy ........ 5
Primary Name server ........ 5
Access Policy Primary – 192.168.41.53 ........ 5
    INPUT POLICY (Default: DROP) ........ 5
    OUTPUT POLICY (Default: DROP) ........ 5
Email Server ........ 6
Access Policy Email – 192.168.41.25 ........ 6
    INPUT POLICY (Default: DROP) ........ 6
    OUTPUT POLICY (Default: DROP) ........ 6
Secondary Name Server ........ 7
Access Policy Secondary Server – 192.168.34.53 ........ 7
    INPUT POLICY (Default: DROP) ........ 7
    OUTPUT POLICY (Default: DROP) ........ 7
Web Server ........ 8
Access Policy Web Server – 192.168.34.80 ........ 8
    INPUT POLICY (Default: DROP) ........ 8
    OUTPUT POLICY (Default: DROP) ........ 9
Host Based Router Access Policy (172.16.41.1) – Primary and Mail Traffic (Default: DROP) ........ 9
    Test Case Validation ........ 10
Host Based Router Access Policy (172.16.34.1) – Secondary and Web traffic (Default: DROP) ........ 15
    Test Case Validation ........ 16
Conclusion ........ 21

## Objective

The objective of this Case Study is to develop a Network based firewall for the routers which route the traffic to the Servers accordingly and drop the other unwanted traffic to the hosts. The main focus of this lab is the FORWARD rule on the network (Router-1 and Router-2).

## Host Machine - 1 Specification

OS – Centos Minimal

Physical Interfaces: eno1 – Link to the internet, eno1:1- 172.16.34.1

Virtual Interface: virbr1 - 192.168.34.0/24

Servers Configured – Web, Secondary DNS

## Host Machine - 2 Specification

OS – Centos Minimal

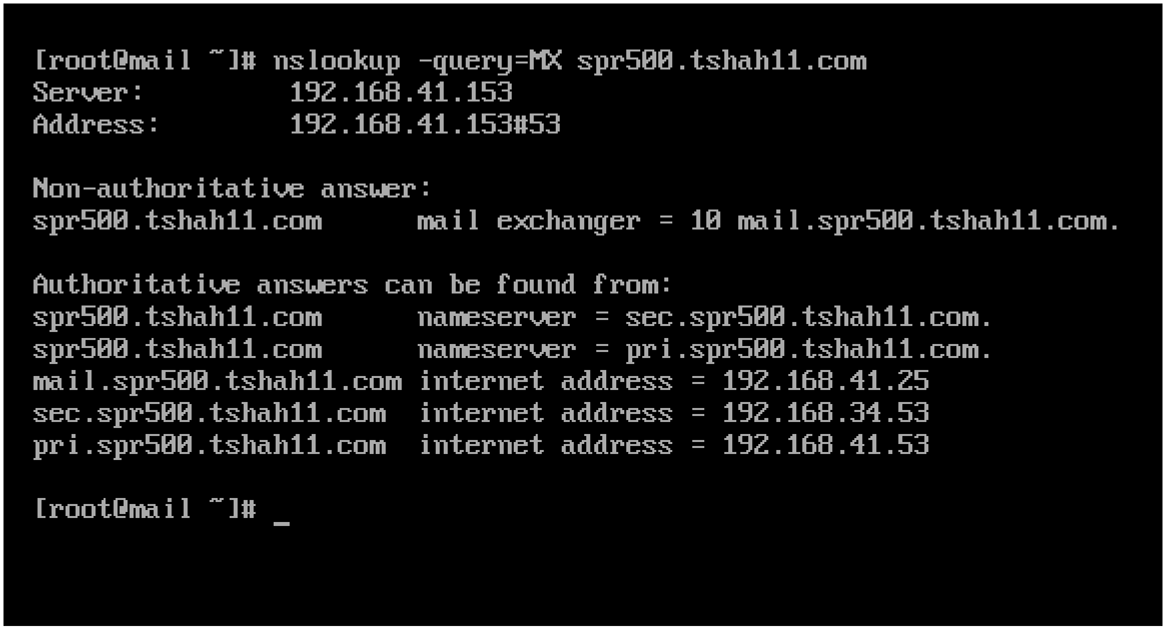
Physical Interfaces: eno1 – Link to the internet, eno1:1- 172.16.41.1

Virtual Interface: virbr3 - 192.168.41.0/24

Servers Configured – DNS, Mail

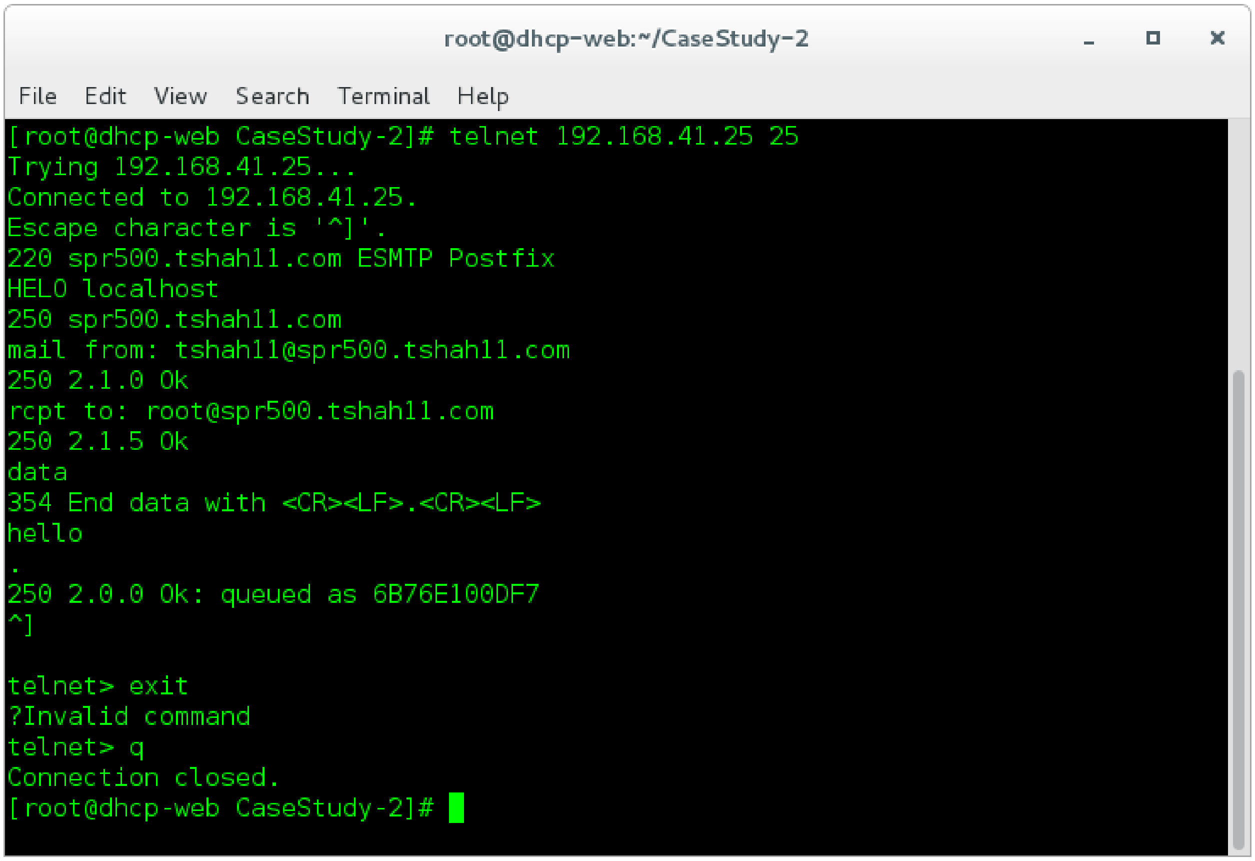
## Proof of Working Servers

### Primary Server & Secondary Server (DNS)



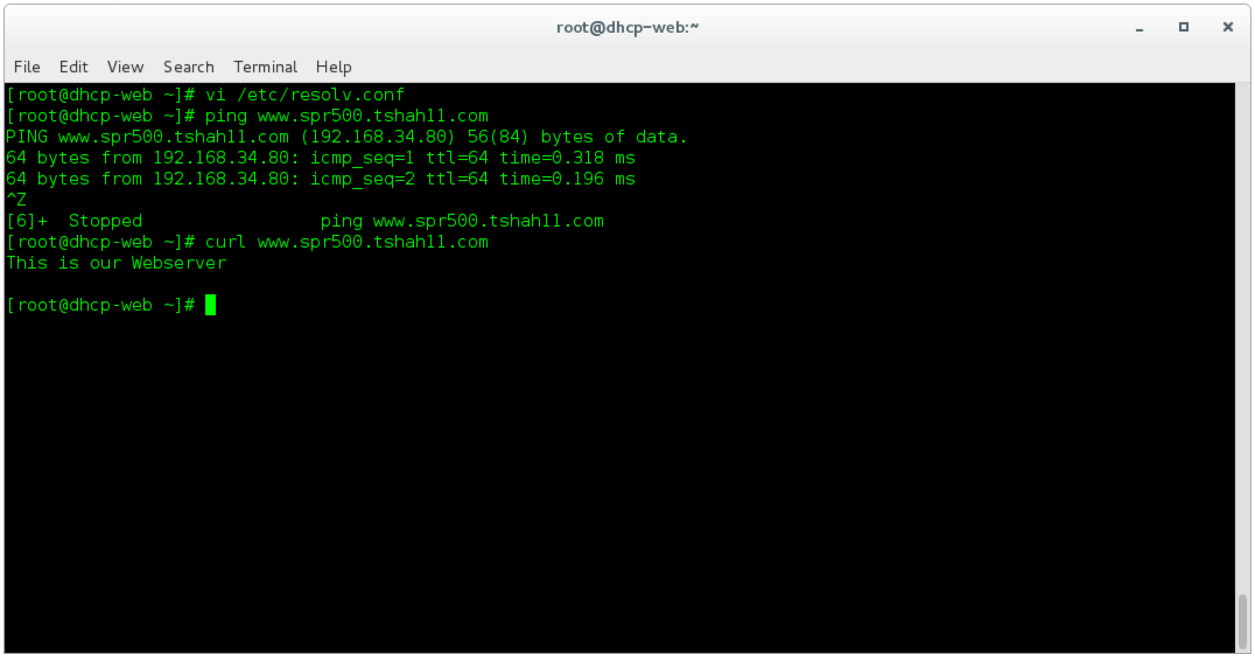
```
[root@mail ~]# nslookup -query=MX spr500.tshah11.com
Server:         192.168.41.153
Address:        192.168.41.153#53

Non-authoritative answer:
spr500.tshah11.com      mail exchanger = 10 mail.spr500.tshah11.com.

Authoritative answers can be found from:
spr500.tshah11.com      nameserver = sec.spr500.tshah11.com.
spr500.tshah11.com      nameserver = pri.spr500.tshah11.com.
mail.spr500.tshah11.com internet address = 192.168.41.25
sec.spr500.tshah11.com  internet address = 192.168.34.53
pri.spr500.tshah11.com  internet address = 192.168.41.53

[root@mail ~]# _
```

## Mail Server


root@dhcp-web:~/CaseStudy-2
File Edit View Search Terminal Help

```
[root@dhcp-web CaseStudy-2]# telnet 192.168.41.25 25
Trying 192.168.41.25...
Connected to 192.168.41.25.
Escape character is '^]'.
220 spr500.tshah11.com ESMTP Postfix
HELO localhost
250 spr500.tshah11.com
mail from: tshah11@spr500.tshah11.com
250 2.1.0 Ok
rcpt to: root@spr500.tshah11.com
250 2.1.5 Ok
data
354 End data with <CR><LF>.<CR><LF>
hello
.
250 2.0.0 Ok: queued as 6B76E100DF7
^]

telnet> exit
?Invalid command
telnet> q
Connection closed.
[root@dhcp-web CaseStudy-2]# 
```

## Working Web Server



```
[root@dhcp-web ~]# vi /etc/resolv.conf
[root@dhcp-web ~]# ping www.spr500.tshah11.com
PING www.spr500.tshah11.com (192.168.34.80) 56(84) bytes of data.
64 bytes from 192.168.34.80: icmp_seq=1 ttl=64 time=0.318 ms
64 bytes from 192.168.34.80: icmp_seq=2 ttl=64 time=0.196 ms
^Z
[6]+  Stopped                 ping www.spr500.tshah11.com
[root@dhcp-web ~]# curl www.spr500.tshah11.com
This is our Webserver

[root@dhcp-web ~]# 
```

# VM Specifications & Access Policy

## Primary Name server

OS – CentOS minimal

Domain – spr500.tshah11.com

IP – 192.168.41.53

## Access Policy Primary – 192.168.41.53

*****R – RELATED, E – ESTABLISHED, N – NEW**

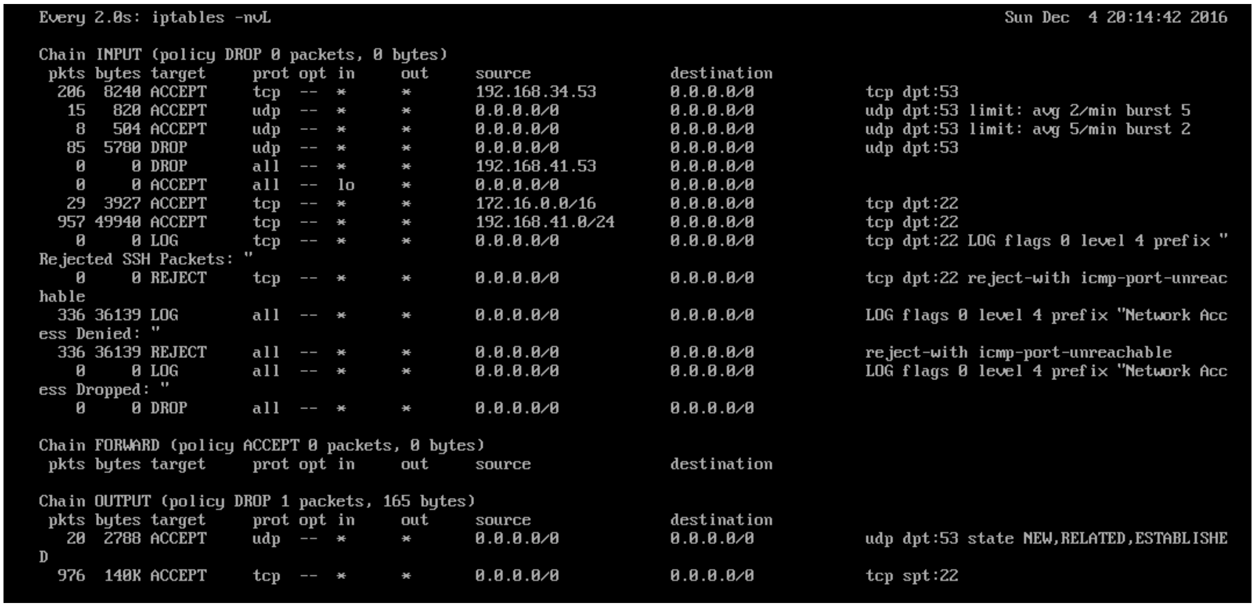
## INPUT POLICY (Default: DROP)

| **Source IP** | **Protocol** | **Destination Port** | **state** | **Extension** | **Access policy** |
|---|---|---|---|---|---|
| * | UDP | 53 | * | Limiter - 7/min | ACCEPT |
| * | UDP | 53 | * | Limiter - 10/min burst(2) | ACCEPT |
| * | UDP | 53 | * | * | DROP |
| 192.168.34.53 | TCP | 53 | * | * | ACCEPT |
| 172.16.0.0/16 | TCP | 22 | NEW | * | ACCEPT,LOG |
| 172.16.0.0/16 | TCP | 22 | R, E | * | ACCEPT |
| 192.168.34.53 | TCP | 53 | * | * | ACCEPT |
| * | TCP | 22 | * | * | LOG |
| * | TCP | 22 | * | * | REJECT |
| * | * | * | * | * | LOG |
| * | * | * | * | * | DROP |

## OUTPUT POLICY (Default: DROP)

| **Destination IP** | **Protocol** | **Source Port** | **state** | **Access policy** |
|---|---|---|---|---|
| * | UDP | 53 | * | ACCEPT |
| * | TCP | 53 | * | ACCEPT |
| * | TCP | 22 | * | ACCEPT |

Primary-vm-fw-stat – **watch iptables –nvL**



```
Every 2.0s: iptables -nvL                                                     Sun Dec  4 20:14:42 2016

Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
  206  8240 ACCEPT     tcp  --  *      *       192.168.34.53        0.0.0.0/0            tcp dpt:53
   15   820 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53 limit: avg 2/min burst 5
    8   504 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53 limit: avg 5/min burst 2
   85  5780 DROP       udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53
    0     0 DROP       all  --  *      *       192.168.41.53        0.0.0.0/0
    0     0 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
   29  3927 ACCEPT     tcp  --  *      *       172.16.0.0/16        0.0.0.0/0            tcp dpt:22
  957 49940 ACCEPT     tcp  --  *      *       192.168.41.0/24      0.0.0.0/0            tcp dpt:22
    0     0 LOG        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:22 LOG flags 0 level 4 prefix "
Rejected SSH Packets: "
    0     0 REJECT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:22 reject-with icmp-port-unreac
hable
  336 36139 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0            LOG flags 0 level 4 prefix "Network Acc
ess Denied: "
  336 36139 REJECT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            reject-with icmp-port-unreachable
    0     0 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0            LOG flags 0 level 4 prefix "Network Acc
ess Dropped: "
    0     0 DROP       all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy DROP 1 packets, 165 bytes)
 pkts bytes target     prot opt in     out     source               destination
   20  2788 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53 state NEW,RELATED,ESTABLISHE
D
  976  140K ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp spt:22
```

## Email Server

OS – CentOS minimal

IP – 192.168.41.25

## Access Policy Email – 192.168.41.25

## INPUT POLICY (Default: DROP)

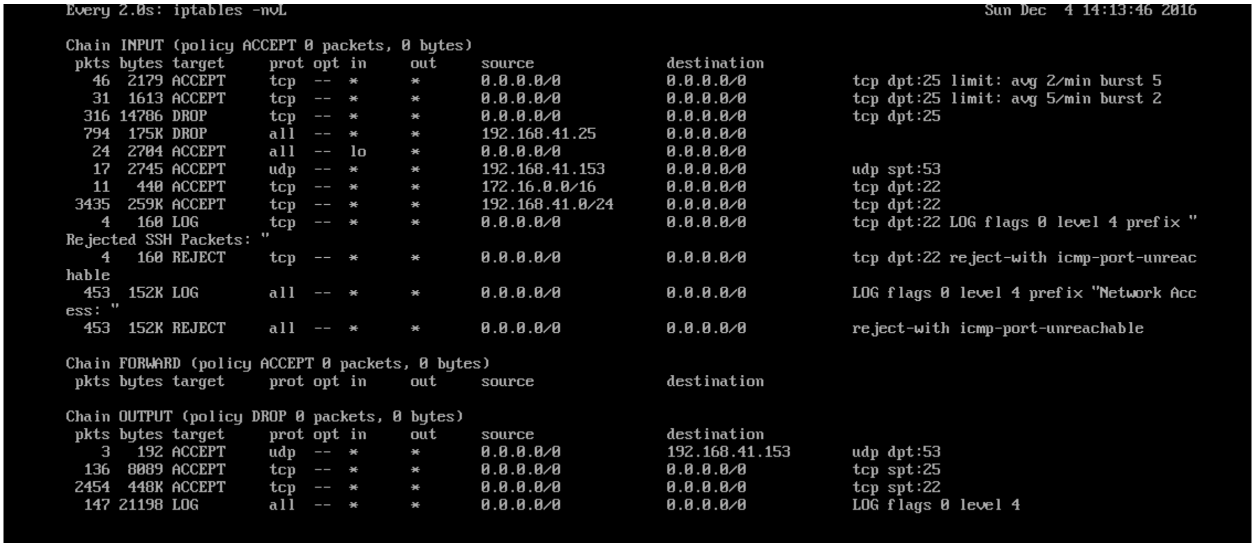
| **Source IP** | **Protocol** | **Destination Port** | **state** | **Extension** | **Access policy** |
|---|---|---|---|---|---|
| * | TCP | 25 | * | Limiter - 7/min | ACCEPT |
| * | TCP | 25 | * | Limiter - 10/min burst(2) | ACCEPT |
| * | TCP | 25 | * | * | DROP |
| 192.168.41.153 | UDP | 53 | * | * | ACCEPT |
| Lo | * | * | * | * | ACCEPT |
| 172.16.0.0/16, 192.168.134.0/24 | TCP | 22 | * | * | ACCEPT |
| * | TCP | 22 | * | * | LOG |
| * | TCP | 22 | * | * | REJECT |
| * | * | * | * | * | LOG |
| * | * | * | * | * | DROP |

## OUTPUT POLICY (Default: DROP)

| **Destination IP** | **Protocol** | **Source Port** | **d-port** | **state** | **Access policy** |
|---|---|---|---|---|---|
| 192.168.41.153 | UDP | * | 53 | * | ACCEPT |

| | | | | | |
|---|---|---|---|---|---|
| * | * | 25 | * | * | ACCEPT |
| 172.16.0.0/16 | TCP | 22 | * | * | ACCEPT |

Mailserver-vm-fw-stat – **watch iptables –nvL**



```
Every 2.0s: iptables -nvL                                                                          Sun Dec  4 14:13:46 2016

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
   46  2179 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:25 limit: avg 2/min burst 5
   31  1613 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:25 limit: avg 5/min burst 2
  316 14786 DROP       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:25
  794  175K DROP       all  --  *      *       192.168.41.25        0.0.0.0/0
   24  2704 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
   17  2745 ACCEPT     udp  --  *      *       192.168.41.153       0.0.0.0/0            udp spt:53
   11   440 ACCEPT     tcp  --  *      *       172.16.0.0/16        0.0.0.0/0            tcp dpt:22
 3435  259K ACCEPT     tcp  --  *      *       192.168.41.0/24      0.0.0.0/0            tcp dpt:22
    4   160 LOG        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:22 LOG flags 0 level 4 prefix "
Rejected SSH Packets: "
    4   160 REJECT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:22 reject-with icmp-port-unreac
hable
  453  152K LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0            LOG flags 0 level 4 prefix "Network Acc
ess: "
  453  152K REJECT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    3   192 ACCEPT     udp  --  *      *       0.0.0.0/0            192.168.41.153       udp dpt:53
  136  8089 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp spt:25
 2454  448K ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp spt:22
  147 21198 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0            LOG flags 0 level 4
```

## Secondary Name Server

OS – CentOS minimal

IP – 192.168.34.53

## Access Policy Secondary Server – 192.168.34.53

## INPUT POLICY (Default: DROP)

| **Source IP** | **Protocol** | **Destination Port** | **s-port** | **state** | **Extension** | **Access policy** |
|---|---|---|---|---|---|---|
| * | UDP | 53 | * | * | Limiter - 7/min | ACCEPT |
| * | UDP | 53 | * | * | Limiter - 10/min burst(2) | ACCEPT |
| * | UDP | 53 | * | * | * | DROP |
| 192.168.41.53 | TCP | * | 53 | * | * | ACCEPT |
| 192.168.41.53 | UDP | * | 53 | * | * | ACCEPT |
| 172.16.0.0/16 | TCP | 22 | | NEW | * | LOG, ACCEPT |
| 172.16.0.0/16 | TCP | 22 | | R,E | * | ACCEPT |
| * | TCP | 22 | | * | * | LOG |
| * | TCP | 22 | | * | * | ACCEPT |
| * | * | * | | * | * | LOG |
| * | * | * | | * | * | DROP |

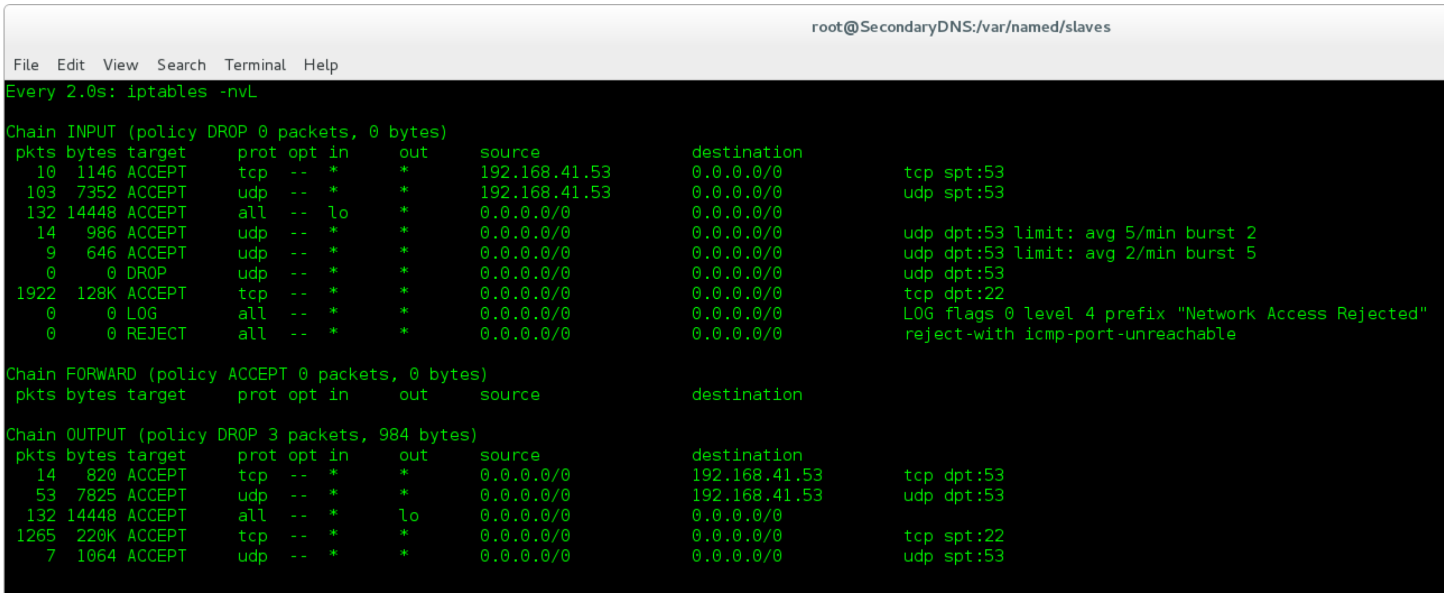
## OUTPUT POLICY (Default: DROP)

| **Destination IP** | **Protocol** | **Source Port** | **d-port** | **State** | **Access policy** |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| 192.168.41.53 | Tcp | * | 53 | * | ACCEPT |
| 192.168.41.53 | UDP | * | 53 | * | ACCEPT |
| Lo | All | * | * | * | ACCEPT |
| * | UDP | 53 | | * | ACCEPT |
| * | TCP | 22 | | * | ACCEPT |

Secondary-vm-fw-stat – **watch iptables –nvL**

****Note – the ssh and Log did not populate because most were dropped by the host itself.**


root@SecondaryDNS:/var/named/slaves
File Edit View Search Terminal Help

```
Every 2.0s: iptables -nvL

Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
   10  1146 ACCEPT     tcp  --  *      *       192.168.41.53        0.0.0.0/0            tcp spt:53
  103  7352 ACCEPT     udp  --  *      *       192.168.41.53        0.0.0.0/0            udp spt:53
  132 14448 ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
   14   986 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53 limit: avg 5/min burst 2
    9   646 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53 limit: avg 2/min burst 5
    0     0 DROP       udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp dpt:53
 1922  128K ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:22
    0     0 LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0            LOG flags 0 level 4 prefix "Network Access Rejected"
    0     0 REJECT     all  --  *      *       0.0.0.0/0            0.0.0.0/0            reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy DROP 3 packets, 984 bytes)
 pkts bytes target     prot opt in     out     source               destination
   14   820 ACCEPT     tcp  --  *      *       0.0.0.0/0            192.168.41.53        tcp dpt:53
   53  7825 ACCEPT     udp  --  *      *       0.0.0.0/0            192.168.41.53        udp dpt:53
  132 14448 ACCEPT     all  --  *      lo      0.0.0.0/0            0.0.0.0/0
 1265  220K ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp spt:22
    7  1064 ACCEPT     udp  --  *      *       0.0.0.0/0            0.0.0.0/0            udp spt:53
```

## Web Server

OS – CentOS minimal

IP – 192.168.34.80

## Access Policy Web Server – 192.168.34.80

## INPUT POLICY (Default: DROP)

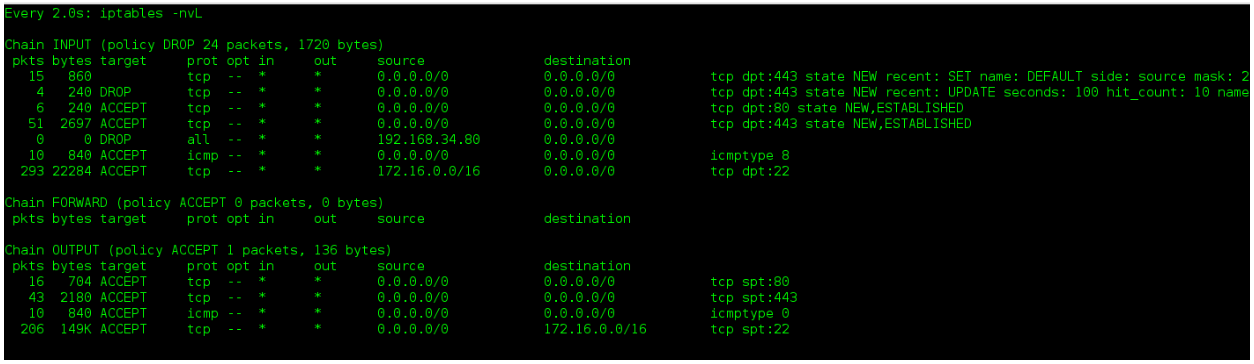
| **Source IP** | **Protocol** | **Destination Port** | **state** | **Extension** | **Access policy** |
|---|---|---|---|---|---|
| * | TCP | 443 | N | Recent source mask – 255.255.255.255 | - |
| * | TCP | 443 | N | hit count 10, update 100 seconds | DROP |
| * | TCP | 80 | N,E | * | ACCEPT |
| * | TCP | 443 | N,E | * | ACCEPT |
| 192.168.34.80 | * | * | * | * | DROP |
| 172.16.0.0/16 | TCP | 22 | R,E | * | ACCEPT |
| 172.16.0.0/16 | TCP | 22 | N | * | ACCEPT, LOG |
| * | ICMP | * | * | Icmp type 8 | ACCEPT |

| | | | | | |
|---|---|---|---|---|---|
| * | * | * | * | * | LOG |
| * | * | * | * | * | DROP |

## OUTPUT POLICY (Default: DROP)

| Destination IP | Protocol | Source Port | State | Access policy |
|---|---|---|---|---|
| * | TCP | 80 | * | ACCEPT |
| * | TCP | 443 | * | ACCEPT |
| * | TCP | 22 | * | ACCEPT |
| 172.16.0.0/16 | ICMP | * | * | ACCEPT |

Webserver-vm-fw-stat – **watch iptables –nvL**



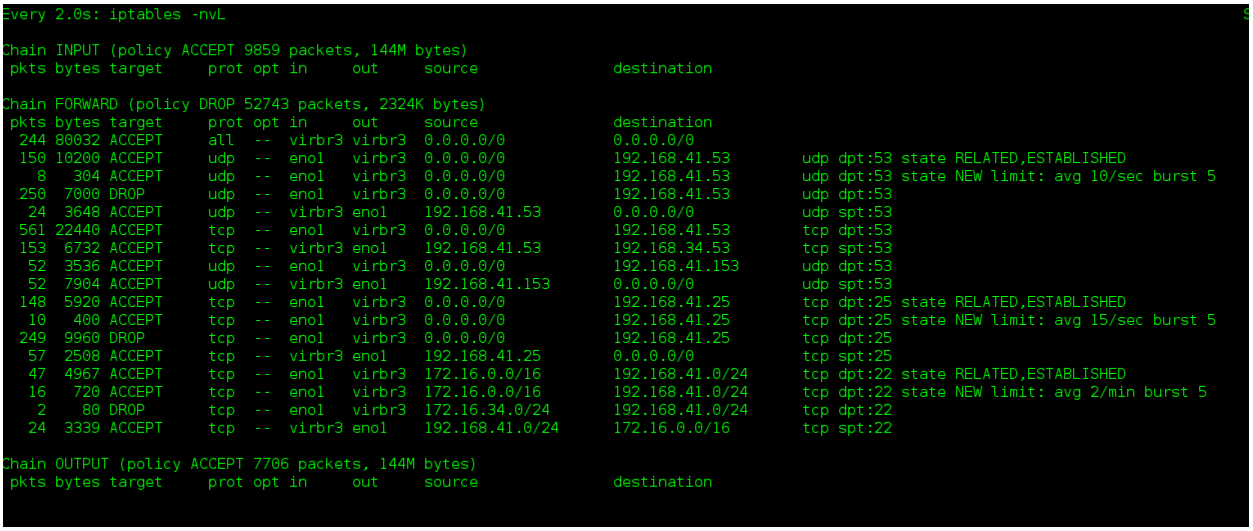
```
Every 2.0s: iptables -nvL

Chain INPUT (policy DROP 24 packets, 1720 bytes)
 pkts bytes target     prot opt in     out     source               destination
   15   860            tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:443 state NEW recent: SET name: DEFAULT side: source mask: 2
    4   240 DROP       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:443 state NEW recent: UPDATE seconds: 100 hit_count: 10 name
    6   240 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:80 state NEW,ESTABLISHED
   51  2697 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp dpt:443 state NEW,ESTABLISHED
    0     0 DROP       all  --  *      *       192.168.34.80        0.0.0.0/0
   10   840 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0            icmptype 8
  293 22284 ACCEPT     tcp  --  *      *       172.16.0.0/16        0.0.0.0/0            tcp dpt:22

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 1 packets, 136 bytes)
 pkts bytes target     prot opt in     out     source               destination
   16   704 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp spt:80
   43  2180 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0            tcp spt:443
   10   840 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0            icmptype 0
  206  149K ACCEPT     tcp  --  *      *       0.0.0.0/0            172.16.0.0/16        tcp spt:22
```

## Host Based Router Access Policy (172.16.41.1) – Primary and Mail Traffic (Default: DROP)

| ID | Source IP | Destination IP | Protocol | Input | Output | S-Port | D-Port | Extension | Access |
|---|---|---|---|---|---|---|---|---|---|
| 1 | * | 192.168.41.53 | UDP | eno1 | virbr3 | * | 53 | State RELATED,ESTABLISHED | ACCEPT |
| | * | 192.168.41.53 | UDP | eno1 | virbr3 | * | 53 | State NEW, limiter 10/sec burst 5 | ACCEPT |
| 2 | * | 192.168.41.53 | UDP | eno1 | virbr3 | * | 53 | | DROP |
| 3 | 192.168.41.53 | * | UDP | virbr3 | eno1 | 53 | * | | ACCEPT |
| 4 | * | 192.168.41.53 | TCP | eno1 | virbr3 | * | 53 | | ACCEPT |
| 5 | 192.168.41.53 | 192.168.34.53 | TCP | virbr3 | eno1 | 53 | * | | ACCEPT |
| 6 | * | 192.168.41.153 | UDP | Eno1 | Virbr3 | * | 53 | | ACCEPT |
| | 192.168.41.153 | * | UDP | Virbr3 | Eno1 | 53 | * | | ACCEPT |
| 7 | * | 192.168.41.25 | TCP | eno1 | virbr3 | * | 25 | State RELATED,ESTABLISHED | ACCEPT |
| | * | 192.168.41.25 | TCP | eno1 | virbr3 | * | 25 | State NEW, limiter 15/sec burst 5 | ACCEPT |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 8 | * | 192.168.41.25 | TCP | eno1 | virbr3 | * | 25 | | DROP |
| 9 | 192.168.41.25 | * | TCP | virbr3 | eno1 | 25 | * | | ACCEPT |
| 10 | 172.16.0.0/24 | 192.168.41.0/24 | TCP | eno1 | virbr3 | * | 22 | State RELATED,ESTABLISHED | ACCEPT |
| | 172.16.0.0/24 | 192.168.41.0/24 | TCP | eno1 | virbr3 | * | 22 | State NEW, limiter 2/min burst 5 | ACCEPT |
| 11 | 172.16.0.0/24 | 192.168.41.0/24 | TCP | eno1 | virbr3 | * | 22 | | DROP |
| 12 | 192.168.41.0/24 | 172.16.0.0/24 | TCP | virbr3 | eno1 | 22 | * | | ACCEPT |
| 13 | * | * | ANY | Virbr3 | Virbr3 | * | * | | ACCEPT |

Host-2-fw-stat – **watch iptables –nvL (Default: DROP)**



```
Every 2.0s: iptables -nvL

Chain INPUT (policy ACCEPT 9859 packets, 144M bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain FORWARD (policy DROP 52743 packets, 2324K bytes)
 pkts bytes target     prot opt in     out     source               destination
  244 80032 ACCEPT     all  --  virbr3 virbr3  0.0.0.0/0            0.0.0.0/0
  150 10200 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state RELATED,ESTABLISHED
    8   304 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state NEW limit: avg 10/sec burst 5
  250  7000 DROP       udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53
   24  3648 ACCEPT     udp  --  virbr3 eno1    192.168.41.53        0.0.0.0/0            udp spt:53
  561 22440 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        tcp dpt:53
  153  6732 ACCEPT     tcp  --  virbr3 eno1    192.168.41.53        192.168.34.53        tcp spt:53
   52  3536 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.153       udp dpt:53
   52  7904 ACCEPT     udp  --  virbr3 eno1    192.168.41.153       0.0.0.0/0            udp spt:53
  148  5920 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state RELATED,ESTABLISHED
   10   400 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state NEW limit: avg 15/sec burst 5
  249  9960 DROP       tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25
   57  2508 ACCEPT     tcp  --  virbr3 eno1    192.168.41.25        0.0.0.0/0            tcp spt:25
   47  4967 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state RELATED,ESTABLISHED
   16   720 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state NEW limit: avg 2/min burst 5
    2    80 DROP       tcp  --  eno1   virbr3  172.16.34.0/24       192.168.41.0/24      tcp dpt:22
   24  3339 ACCEPT     tcp  --  virbr3 eno1    192.168.41.0/24      172.16.0.0/16        tcp spt:22

Chain OUTPUT (policy ACCEPT 7706 packets, 144M bytes)
 pkts bytes target     prot opt in     out     source               destination
```

## Test Case Validation

### Test Case #1

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all the DNS traffic which has the destination as the Primary Name Server

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p udp –d 192.168.41.53 –i eno1 –o virbr3 –dport 53 –m state –state RELATED, ESTABLISHED –j ACCEPT
iptables –A FORWARD –p udp –d 192.168.41.53 –i eno1 –o virbr3 –dport 53 –m state –state NEW –m limit –limit 10/sec burst 5 –j ACCEPT
```

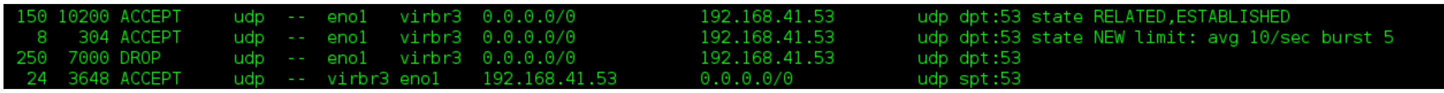
**Expected result**

The router should forward the requests to the DNS server and get a response.

**Actual Result**

We got the query answer from the DNS server

**Proof**



```
150 10200 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state RELATED,ESTABLISHED
  8   304 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state NEW limit: avg 10/sec burst 5
250  7000 DROP       udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53
 24  3648 ACCEPT     udp  --  virbr3 eno1    192.168.41.53        0.0.0.0/0            udp spt:53
```

## Test Case #2

**Description**

Net filter rules on host router to drop DNS traffic and forward accordingly.

**Purpose**

To drop all the DNS traffic which has the destination as the Primary Name Server and exceeds the packet limit.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p udp –d 192.168.41.53 –i eno1 –o virbr3 –dport 53 –j DROP
```

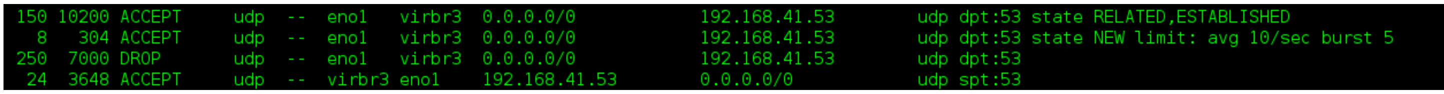
**Expected result**

The router should drop the requests to the DNS server.

**Actual Result**

We got the query answer from the DNS server

**Proof**



```
150 10200 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state RELATED,ESTABLISHED
  8   304 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state NEW limit: avg 10/sec burst 5
250  7000 DROP       udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53
 24  3648 ACCEPT     udp  --  virbr3 eno1    192.168.41.53        0.0.0.0/0            udp spt:53
```

## Test Case #3

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all the DNS traffic which has the destination as the Primary Name Server and send a response back.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p udp –s 192.168.41.53 –o eno1 –i virbr3 –sport 53 –j ACCEPT
```

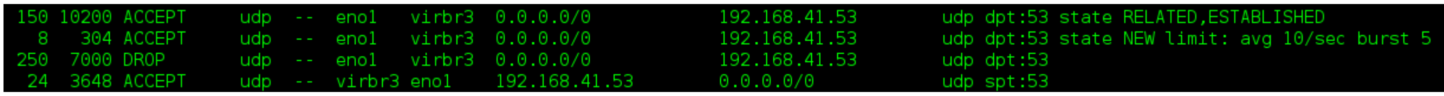
**Expected result**

The router should drop the requests to the DNS server.

**Actual Result**

We did not get query answer from the DNS server

**Proof**



```
150 10200 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state RELATED,ESTABLISHED
  8   304 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53 state NEW limit: avg 10/sec burst 5
250  7000 DROP       udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        udp dpt:53
 24  3648 ACCEPT     udp  --  virbr3 eno1    192.168.41.53        0.0.0.0/0            udp spt:53
```

## Test Case #4

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all the DNS traffic for zone transfer which has the destination as the Primary Name Server and is TCP traffic.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –d 192.168.41.53 –i eno1 –o virbr3 –dport 53 –j ACCEPT
```

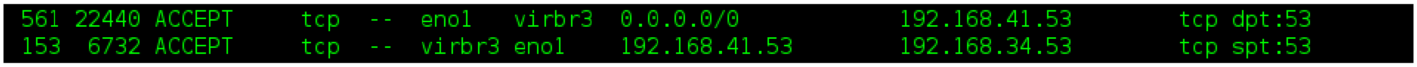
**Expected result**

The router should drop the requests to the DNS server.

**Actual Result**

We got the query answer from the DNS server

**Proof**



```
 561 22440 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        tcp dpt:53
 153  6732 ACCEPT     tcp  --  virbr3 eno1    192.168.41.53        192.168.34.53        tcp spt:53
```

**Test Case #5**

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all the DNS traffic which has the destination as the Primary Name Server and the source of the Secondary Name Server and vice versa in order to send and accept master-slave replication traffic.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –d 192.168.34.53 –s 192.168.41.53 –o eno1 –i virbr3 –sport 53 –j
ACCEPT
```

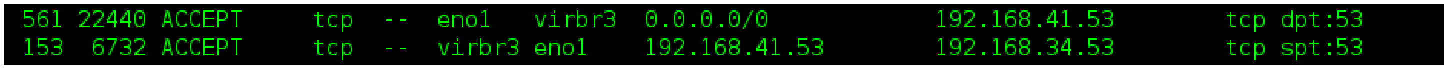
**Expected result**

The router should forward the requests to the Primary DNS server and get a response to send back to the Secondary Server. The Secondary Server should have its zone files populated.

**Actual Result**

The Secondary Server receives the transfer and has a populated zone file.

**Proof**



```
 561 22440 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.53        tcp dpt:53
 153  6732 ACCEPT     tcp  --  virbr3 eno1    192.168.41.53        192.168.34.53        tcp spt:53
```

**Test Case #6**

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all the DNS traffic which has the destination as the Primary Name Server and source of the caching server and vice versa.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –d 192.168.41.153 –i eno1 –o virbr3 –dport 53 –j ACCEPT
iptables –A FORWARD –p tcp –s 192.168.41.153 –o eno1 –i virbr3 –sport 53 –j ACCEPT
```

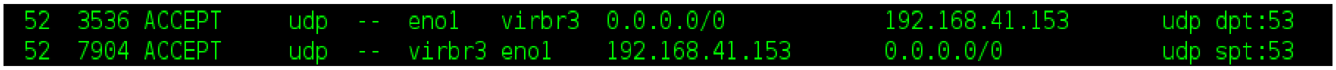
**Expected result**

The router should accept the requests to the DNS server.

**Actual Result**

We got the query answer from the caching DNS server

**Proof**



```
 52  3536 ACCEPT     udp  --  eno1   virbr3  0.0.0.0/0            192.168.41.153       udp dpt:53
 52  7904 ACCEPT     udp  --  virbr3 eno1    192.168.41.153       0.0.0.0/0            udp spt:53
```

## Test Case #7

**Description**

Net filter rules on host router to accept mail traffic and forward accordingly.

**Purpose**

To accept all mail traffic destined to go to the mail server

**Test environment**

Mail traffic will be sent using scapy. Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –d 192.168.41.25 –i eno1 –o virbr3 –dport 53 –m state –state
RELATED,ESTABLISHED –j ACCEPT
iptables –A FORWARD –p tcp –d 192.168.41.25 –i eno1 –o virbr3 –dport 53 –m state –state NEW –m
limit –limit 10/sec burst 5 –j ACCEPT
```

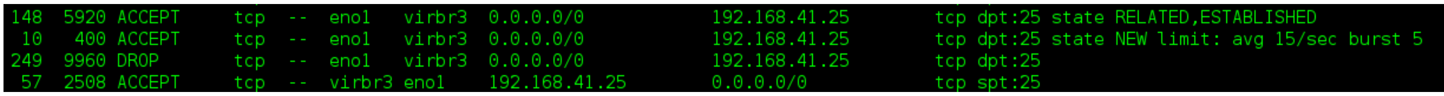
**Expected result**

The mail traffic is accepted and packets are being sent through on both of the forward chain rules.

**Actual Result**

Mail traffic is accepted and the packets were allowed and sent through the corresponding rules.

**Proof**



```
148  5920 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state RELATED,ESTABLISHED
 10   400 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state NEW limit: avg 15/sec burst 5
249  9960 DROP       tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25
 57  2508 ACCEPT     tcp  --  virbr3 eno1    192.168.41.25        0.0.0.0/0            tcp spt:25
```

## Test Case #8

**Description**

Net filter rules on host router to drop DNS traffic and forward accordingly.

**Purpose**

To drop all the mail traffic which has the destination as the Mail Server and exceeds the packet limit.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –d 192.168.41.25 –i eno1 –o virbr3 –dport 25 –j DROP
```

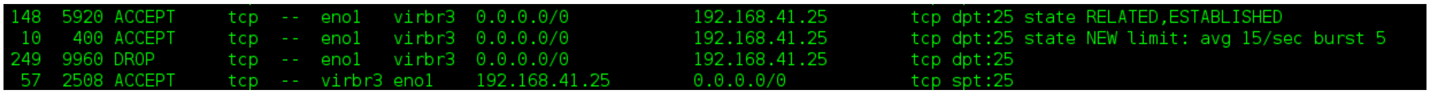
**Expected result**

The router should drop the requests to the mail server.

**Actual Result**

We got the no response from the mail server.

**Proof**



```
148  5920 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state RELATED,ESTABLISHED
 10   400 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state NEW limit: avg 15/sec burst 5
249  9960 DROP       tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25
 57  2508 ACCEPT     tcp  --  virbr3 eno1    192.168.41.25        0.0.0.0/0            tcp spt:25
```

## Test Case #9

**Description**

Net filter rules on host router to accept Mail traffic and forward accordingly.

**Purpose**

To accept all the Mail traffic which has the destination as the Primary Name Server and send a response back.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –s 192.168.41.25 –o eno1 –i virbr3 –sport 25 –j ACCEPT
```

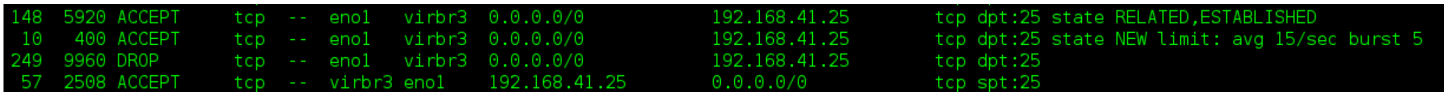
**Expected result**

The router should drop the requests to the Mail server.

**Actual Result**

We did not get query answer from the Mail server

**Proof**



```
148  5920 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state RELATED,ESTABLISHED
 10   400 ACCEPT     tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25 state NEW limit: avg 15/sec burst 5
249  9960 DROP       tcp  --  eno1   virbr3  0.0.0.0/0            192.168.41.25        tcp dpt:25
 57  2508 ACCEPT     tcp  --  virbr3 eno1    192.168.41.25        0.0.0.0/0            tcp spt:25
```

## Test Case #10

**Description**

Net filter rules on host router to accept SSH traffic and forward accordingly.

**Purpose**

To accept all SSH traffic destined to go to the 192.168.41.0/24 network

**Test environment**

Mail traffic will be sent using scapy. Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –s 172.16.0.0/16 –d 192.168.41.0/24 –i eno1 –o virbr3 –dport 22 –m state –state RELATED,ESTABLISHED –j ACCEPT
iptables –A FORWARD –p tcp –s 172.16.0.0/16 –d 192.168.41.0/24 –i eno1 –o virbr3 –dport 22 –m state –state NEW –m limit –limit 10/sec burst 5 –j ACCEPT
```

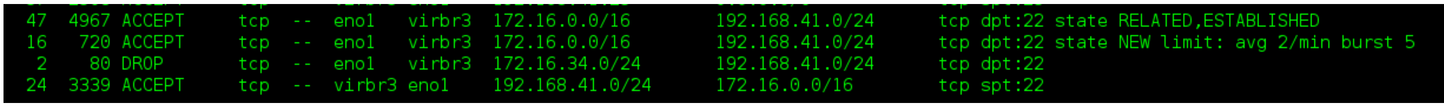
**Expected result**

The SSH traffic is accepted and packets are being sent through on both of the forward chain rules.

**Actual Result**

SSH traffic is accepted and the packets were allowed and sent through the corresponding rules.

**Proof**



```
 47  4967 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state RELATED,ESTABLISHED
 16   720 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state NEW limit: avg 2/min burst 5
  2    80 DROP       tcp  --  eno1   virbr3  172.16.34.0/24       192.168.41.0/24      tcp dpt:22
 24  3339 ACCEPT     tcp  --  virbr3 eno1    192.168.41.0/24      172.16.0.0/16        tcp spt:22
```

## Test Case #11

**Description**

Net filter rules on host router to drop SSH traffic and forward accordingly.

**Purpose**

To drop all the SSH traffic which has the destination for the 192.168.41.0/24 network and exceeds the packet limit.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p tcp –s 172.16.0.0/16 –d 192.168.41.0/24 –i eno1 –o virbr3 –dport 22 –j DROP
```

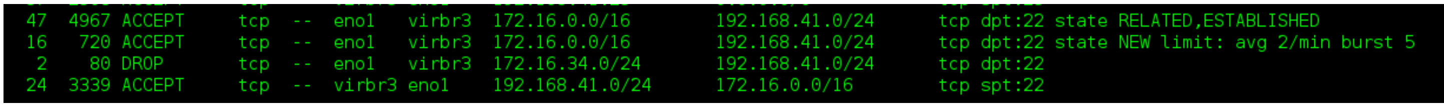
**Expected result**

The router should drop the requests for SSH.

**Actual Result**

The router drops requests.

**Proof**



```
 47  4967 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state RELATED,ESTABLISHED
 16   720 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state NEW limit: avg 2/min burst 5
  2    80 DROP       tcp  --  eno1   virbr3  172.16.34.0/24       192.168.41.0/24      tcp dpt:22
 24  3339 ACCEPT     tcp  --  virbr3 eno1    192.168.41.0/24      172.16.0.0/16        tcp spt:22
```

### Test Case #12

**Description**

Net filter rules on host router to accept SSH traffic and forward accordingly.

**Purpose**

To accept all the SSH traffic which has the destination as the 172.16.0.0/16 network

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD –p udp –s 192.168.41.0/24 –d 172.16.0.0/16 –o eno1 –i virbr3 –sport 22 –j
ACCEPT
```

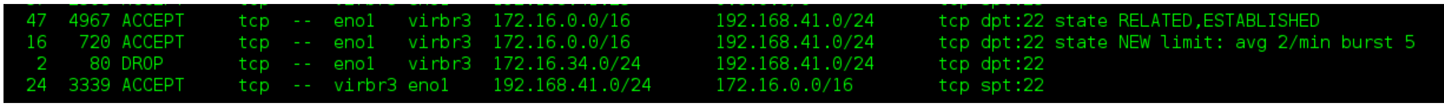
**Expected result**

The router should accept all traffic designated for the 172.16.0.0/16 network.

**Actual Result**

The router forwards the traffic.

**Proof**



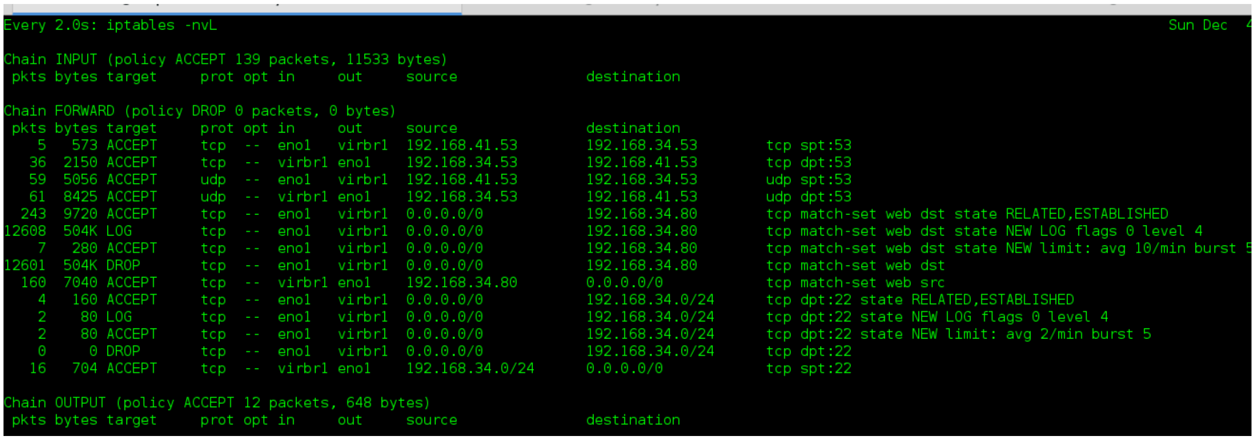
```
 47  4967 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state RELATED,ESTABLISHED
 16   720 ACCEPT     tcp  --  eno1   virbr3  172.16.0.0/16        192.168.41.0/24      tcp dpt:22 state NEW limit: avg 2/min burst 5
  2    80 DROP       tcp  --  eno1   virbr3  172.16.34.0/24       192.168.41.0/24      tcp dpt:22
 24  3339 ACCEPT     tcp  --  virbr3 eno1    192.168.41.0/24      172.16.0.0/16        tcp spt:22
```

## Host Based Router Access Policy (172.16.34.1) – Secondary and Web traffic (Default: DROP)

| ID | Source IP | Destination IP | Protocol | Input | Output | S-Port | D-Port | Extension | Access |
|---|---|---|---|---|---|---|---|---|---|
| 1 | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 80 | State NEW | LOG |
|  | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 80 | State NEW, Limiter – 10/min | ACCEPT |
|  | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 443 | State NEW | LOG |
|  | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 443 | State NEW, Limiter – 10/min | ACCEPT |
| 2 | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 80 | State RELATED/ESTABLISHED | ACCEPT |
|  | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 443 | State RELATED/ESTABLISHED | ACCEPT |
| 3 | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 80 |  | DROP |
|  | * | 192.168.34.80 | TCP | eno1 | Virbr1 | * | 443 |  | DROP |
| 4 | 192.168.34.80 | * | TCP | Virbr1 | eno1 | 80 | * |  | ACCEPT |
|  | 192.168.34.80 | * | TCP | Virbr1 | eno1 | 443 | * |  | ACCEPT |
| 5 | * | 192.168.34.53 | UDP | eno1 | Virbr1 | * | 53 | State NEW | LOG |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | * | 192.168.34.53 | UDP | eno1 | Virbr1 | * | 53 | State NEW | ACCEPT |
| 6 | * | 192.168.34.53 | UDP | eno1 | Virbr1 | * | 53 | State RELATED/ESTABLISHED, Limiter – 5/min | ACCEPT |
| 7 | * | 192.168.34.53 | UDP | eno1 | Virbr1 | * | 53 | | DROP |
| 8 | 192.168.34.53 | * | UDP | Virbr1 | eno1 | 53 | * | | ACCEPT |
| 9 | 172.16.0.0/16 | 192.168.34.0/24 | TCP | eno1 | Virbr1 | * | 22 | State NEW | LOG |
| 10 | 172.16.0.0/16 | 192.168.34.0/24 | TCP | eno1 | Virbr1 | * | 22 | | ACCEPT |
| | 192.168.34.0/24 | 172.16.0.0/26 | TCP | Virbr1 | eno1 | 22 | * | | ACCEPT |

Host-1-fw stat – **watch iptables –nvL**



```
Every 2.0s: iptables -nvL                                                                                Sun Dec  4

Chain INPUT (policy ACCEPT 139 packets, 11533 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    5   573 ACCEPT     tcp  --  eno1   virbr1  192.168.41.53        192.168.34.53        tcp spt:53
   36  2150 ACCEPT     tcp  --  virbr1 eno1    192.168.34.53        192.168.41.53        tcp dpt:53
   59  5056 ACCEPT     udp  --  eno1   virbr1  192.168.41.53        192.168.34.53        udp spt:53
   61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
  243  9720 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state RELATED,ESTABLISHED
12608  504K LOG        tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW LOG flags 0 level 4
    7   280 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW limit: avg 10/min burst 5
12601  504K DROP       tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst
  160  7040 ACCEPT     tcp  --  virbr1 eno1    192.168.34.80        0.0.0.0/0            tcp match-set web src
    4   160 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state RELATED,ESTABLISHED
    2    80 LOG        tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state NEW LOG flags 0 level 4
    2    80 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state NEW limit: avg 2/min burst 5
    0     0 DROP       tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22
   16   704 ACCEPT     tcp  --  virbr1 eno1    192.168.34.0/24      0.0.0.0/0            tcp spt:22

Chain OUTPUT (policy ACCEPT 12 packets, 648 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

## Test Case Validation

### Test Case # 1

**Description**

Net filter rules on host router to accept Web traffic and forward accordingly.

**Purpose**

To log and accept all the new Web traffic which has the destination as the Web Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables –A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 80 -m state --state NEW -j LOG
iptables –A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 80 -m state --state NEW -m limit --limit 10/min -j ACCEPT
iptables -A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 443 -m state --state NEW -j LOG
iptables -A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 443 -m state --state
NEW -m limit --limit 10/min -j ACCEPT
```

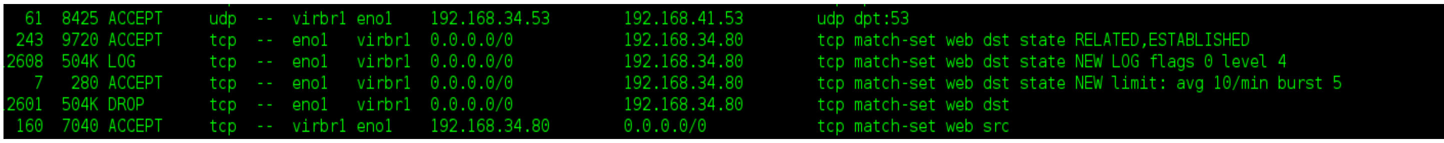
**Expected result**

The client should be able to view the web page using both http and https connections.

**Actual Result**

The client was able to view the web page using both http and https connections.

**Proof**



```
  61  8425 ACCEPT     udp  --  virbr1 enol    192.168.34.53        192.168.41.53        udp dpt:53
 243  9720 ACCEPT     tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state RELATED,ESTABLISHED
2608  504K LOG        tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW LOG flags 0 level 4
   7   280 ACCEPT     tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW limit: avg 10/min burst 5
2601  504K DROP       tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst
 160  7040 ACCEPT     tcp  --  virbr1 enol    192.168.34.80        0.0.0.0/0            tcp match-set web src
```

## Test Case #2

**Description**

Net filter rules on host router to accept Web traffic and forward accordingly.

**Purpose**

To accept related/established Web traffic which has the destination as the Web Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables - A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 80 -m state --state
RELATED, ESTABLISHED -j ACCEPT
iptables -A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 443 -m state --state
RELATED,ESTABLISHED -j ACCEPT
```

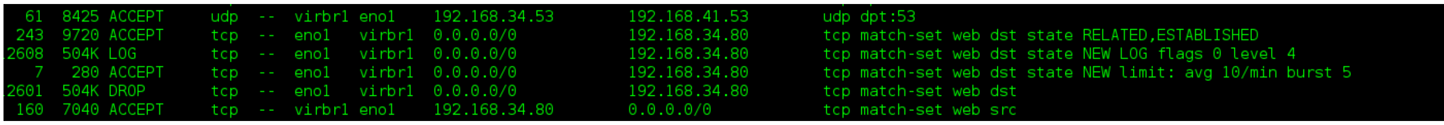
**Expected result**

The client should be able to view the web page using both http and https connections.

**Actual Result**

The client was able to view the web page using both http and https connections.

**Proof**



```
  61  8425 ACCEPT     udp  --  virbr1 enol    192.168.34.53        192.168.41.53        udp dpt:53
 243  9720 ACCEPT     tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state RELATED,ESTABLISHED
2608  504K LOG        tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW LOG flags 0 level 4
   7   280 ACCEPT     tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW limit: avg 10/min burst 5
2601  504K DROP       tcp  --  enol   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst
 160  7040 ACCEPT     tcp  --  virbr1 enol    192.168.34.80        0.0.0.0/0            tcp match-set web src
```

## Test Case #3

**Description**

Net filter rules on host router to drop Web traffic and forward accordingly.

**Purpose**

To drop all other Web traffic that exceeds the limit of 10 connections and has the destination as the Web Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 80 -j DROP
iptables -A FORWARD -d 192.168.34.80/32 -i eno1 -o virbr1 -p tcp -m tcp --dport 443 -j DROP
```

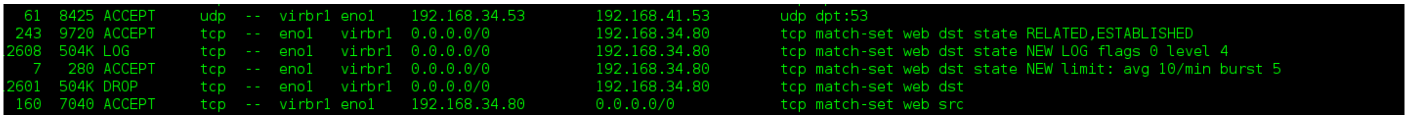
**Expected result**

The client should not be able to view the web page using both http and https connections.

**Actual Result**

The client was not able to view the web page using both http and https connections.

**Proof**



```
  61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
 243  9720 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state RELATED,ESTABLISHED
2608  504K LOG        tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW LOG flags 0 level 4
   7   280 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW limit: avg 10/min burst 5
2601  504K DROP       tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst
 160  7040 ACCEPT     tcp  --  virbr1 eno1    192.168.34.80        0.0.0.0/0            tcp match-set web src
```

## Test Case #4

**Description**

Net filter rules on host router to accept outbound Web traffic and forward accordingly.

**Purpose**

To accept all Web traffic that has the source as the Web Server and is going any destination.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -s 192.168.34.80/32 -i virbr1 -o eno1 -p tcp -m tcp --sport 80 -j ACCEPT
iptables -A FORWARD -s 192.168.34.80/32 -i virbr1 -o eno1 -p tcp -m tcp --sport 443 -j ACCEPT
```

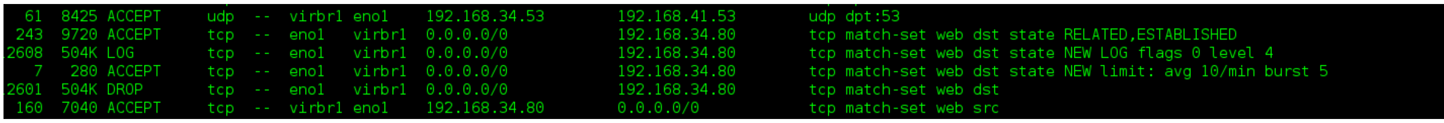
**Expected result**

The client should be able to view the web page using both http and https connections.

**Actual Result**

The client was able to view the web page using both http and https connections.

**Proof**



```
  61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
 243  9720 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state RELATED,ESTABLISHED
2608  504K LOG        tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW LOG flags 0 level 4
   7   280 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst state NEW limit: avg 10/min burst 5
2601  504K DROP       tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.80        tcp match-set web dst
 160  7040 ACCEPT     tcp  --  virbr1 eno1    192.168.34.80        0.0.0.0/0            tcp match-set web src
```

## Test Case #5

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To log and accept all the new DNS traffic which has the destination as the Secondary Name Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -d 192.168.34.53/32 -i eno1 -o virbr1 -p udp -m udp --dport 53 -m state --state
NEW -j LOG
-A FORWARD -d 192.168.34.53/32 -i eno1 -o virbr1 -p udp -m udp --dport 53 -m state --state NEW -j
ACCEPT
```

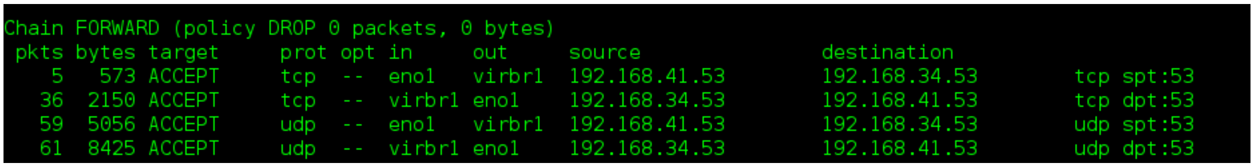
**Expected result**

The router should forward the requests to the DNS server

**Actual Result**

The router forwards the requests and the corresponding iptables rules have packets being sent through the corresponding rules

**Proof**



```
Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    5   573 ACCEPT     tcp  --  eno1   virbr1  192.168.41.53        192.168.34.53        tcp spt:53
   36  2150 ACCEPT     tcp  --  virbr1 eno1    192.168.34.53        192.168.41.53        tcp dpt:53
   59  5056 ACCEPT     udp  --  eno1   virbr1  192.168.41.53        192.168.34.53        udp spt:53
   61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
```

## Test Case #6

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all the related/established DNS traffic which has the destination as the Secondary Name Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -d 192.168.34.53/32 -i eno1 -o virbr1 -p udp -m udp --dport 53 -m state --state RELATED,ESTABLISHED -m limit --limit 5/min -j ACCEPT
```

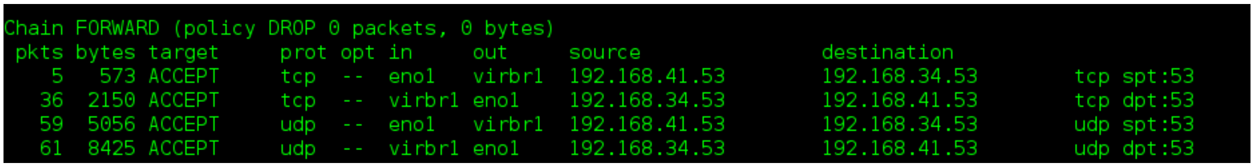
**Expected result**

The router should forward the requests to the DNS server

**Actual Result**

The router forwards the requests and the corresponding iptables rules have packets being sent through the corresponding rules

**Proof**



```
Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    5   573 ACCEPT     tcp  --  eno1   virbr1  192.168.41.53        192.168.34.53        tcp spt:53
   36  2150 ACCEPT     tcp  --  virbr1 eno1    192.168.34.53        192.168.41.53        tcp dpt:53
   59  5056 ACCEPT     udp  --  eno1   virbr1  192.168.41.53        192.168.34.53        udp spt:53
   61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
```

## Test Case #7

**Description**

Net filter rules on host router to drop DNS traffic and forward accordingly.

**Purpose**

To drop all other DNS traffic which exceeds the limit of 5 packets per minute and has the destination as the Secondary Name Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -d 192.168.34.53/32 -i eno1 -o virbr1 -p udp -m udp --dport 53 -j DROP
```

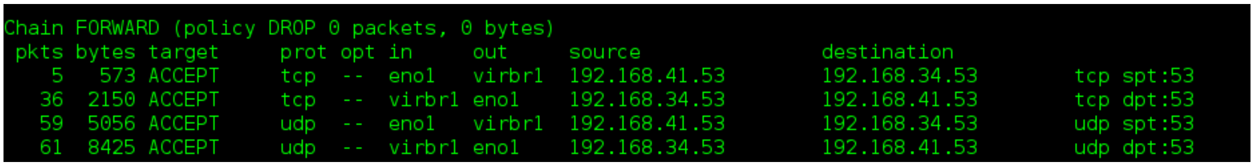
**Expected result**

The router should drop the requests to the DNS server

**Actual Result**

The router drops the requests and the corresponding iptables rules have packets being sent through the corresponding rules

**Proof**



```
Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    5   573 ACCEPT     tcp  --  eno1   virbr1  192.168.41.53        192.168.34.53        tcp spt:53
   36  2150 ACCEPT     tcp  --  virbr1 eno1    192.168.34.53        192.168.41.53        tcp dpt:53
   59  5056 ACCEPT     udp  --  eno1   virbr1  192.168.41.53        192.168.34.53        udp spt:53
   61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
```

### Test Case #8

**Description**

Net filter rules on host router to accept DNS traffic and forward accordingly.

**Purpose**

To accept all outgoing DNS traffic which has the source as the Secondary Name Server.

**Test environment**

Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -s 192.168.34.53/32 -i virbr1 -o eno1 -p udp -m udp --sport 53 -j ACCEPT
```

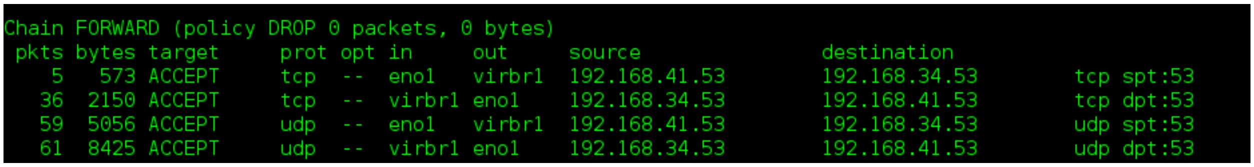
**Expected result**

The router should accept the requests to the DNS server and the client should receive an answer to their query

**Actual Result**

The router accepts the requests and forwards the answers through the rule to the client

**Proof**



```
Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    5   573 ACCEPT     tcp  --  eno1   virbr1  192.168.41.53        192.168.34.53        tcp spt:53
   36  2150 ACCEPT     tcp  --  virbr1 eno1    192.168.34.53        192.168.41.53        tcp dpt:53
   59  5056 ACCEPT     udp  --  eno1   virbr1  192.168.41.53        192.168.34.53        udp spt:53
   61  8425 ACCEPT     udp  --  virbr1 eno1    192.168.34.53        192.168.41.53        udp dpt:53
```

### Test Case #9

**Description**

Net filter rules on host router to log SSH traffic and forward accordingly.

**Purpose**

To log all the SSH traffic from source network 192.168.134.0/24 which has the destination for any of the machines in the 192.168.34.x network and allow remote access to any of these designated machines.

**Test environment**

SSH traffic will be sent using scappy. Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
iptables -A FORWARD -d 192.168.34.0/24 –s 192.168.34.0/24 -i eno1 -o virbr1 -p tcp -m tcp --dport
22 -m state --state NEW -j LOG
```

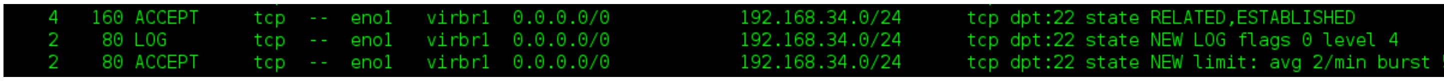
**Expected result**

SSH traffic will be accepted and the packets will be sent through the forward chain rules.

**Actual Result**

SSH traffic is accepted and the packets were allowed and sent through the corresponding rules.

**Proof**



```
    4   160 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state RELATED,ESTABLISHED
    2    80 LOG        tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state NEW LOG flags 0 level 4
    2    80 ACCEPT     tcp  --  eno1   virbr1  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state NEW limit: avg 2/min burst
```

### Test Case #10

**Description**

Net filter rules on host router to accept SSH traffic and forward accordingly.

**Purpose**

To accept all the SSH traffic from source network 192.168.134.0/24 which has the destination for any of the machines in the 192.168.34.x network and allow remote access to any of these designated machines.

**Test environment**

SSH traffic will be sent using scapy. Watch iptables, the logs and the tcpdump on the host to see if the packets are going through.

**Input data**

```
Iptables  A FORWARD -d 192.168.34.0/24 –s 192.168.34.0/24 -i eno1 -o virbr1 -p tcp -m tcp --dport 22  -j ACCEPT
iptables -A FORWARD -s 192.168.34.0/24 –d 192.168.134.0/24 -i virbr1 -o eno1 -p tcp -m tcp --sport 22 -j ACCEPT
```

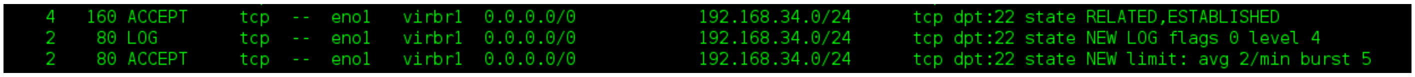
**Expected result**

SSH traffic will be accepted and the packets will be sent through the forward chain rules.

**Actual Result**

SSH traffic is accepted and the packets were allowed and sent through the corresponding rules.

**Proof**



```
    4   160 ACCEPT     tcp  --  enol   virbrl  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state RELATED,ESTABLISHED
    2    80 LOG        tcp  --  enol   virbrl  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state NEW LOG flags 0 level 4
    2    80 ACCEPT     tcp  --  enol   virbrl  0.0.0.0/0            192.168.34.0/24      tcp dpt:22 state NEW limit: avg 2/min burst 5
```

## Conclusion

This CaseStudy was focused on Network-based firewall, we have learned to develop access policy and also implement them in a very secure way as possible.