Linux Standalone Firewall

COMP 8006
Assignment 2

Ian Lee and Luke Tao

Table of Contents

Introduction

The purpose of this assignment was to design, implement and test a standalone Linux firewall, using a packet filter shell script. The following firewall rules are as follows:

- Set the initial default policies.
- Get user specified parameters and create a set of rules that will implement the firewall requirements. Refer to the “Firewall Rules” section of the report.

There are two shell scripts that will test both the internal and external hosts and each shell script has its own set of test cases. After running a test script, the each test case results will be printed to a separate file. For example, test case 1 results for internal hosts will be written under “test1.txt” in the “internal_tests” directory.

Since the machines are equipped with two network cards, one of the machines in the lab will act as a standalone firewall that will have the firewall script. That standalone firewall machine will have one network card (em1) acting as a gateway and will forward datagrams to hosts on its internal hosts on the second network card (p3p1) and vice versa. For more details, refer to the “Design” section of the report.

To make setup easier, two scripts can be used to configure the networks on each of the computers.

Firewall Rules

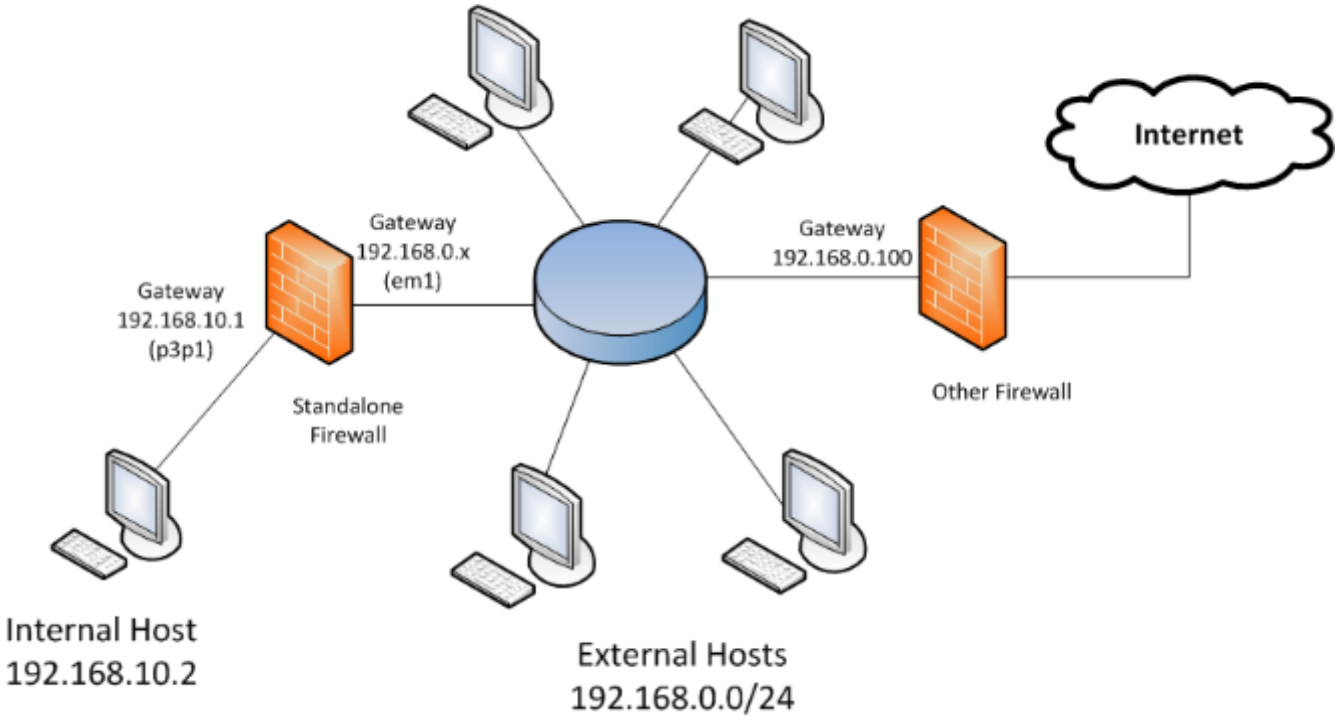
- Inbound/Outbound TCP packets on allowed ports.
- Inbound/Outbound UDP packets on allowed ports.
- Inbound/Outbound ICMP packets based on type numbers.
- All packets that fall through to the default rule will be dropped.
- Drop all packets destined for the firewall host from the outside.
- Do not accept any packets with a source address from the outside matching your internal network.
- You must ensure the you reject those connections that are coming the “wrong” way (i.e., inbound SYN packets to high ports).
- Accept fragments.
- Accept all TCP packets that belong to an existing connection (on allowed ports).
- Drop all TCP packets with the SYN and FIN bit set.
- Do not allow Telnet packets at all.
- Block all external traffic directed to ports 32768 - 32775, 137 – 139, TCP ports 111 and 515.
- For FTP and SSH services, set control connections to "Minimum Delay" and FTP data to "Maximum Throughput".

Constraints

- Firewall/packet filter must be designed and implemented using **Netfilter**.
- Must have two sections:
    - User Configurable
    - Implementation
- User configuration section that permits the user to set the following parameters:
    - External network interface
    - Internal network interface
    - TCP services both inbound and outbound
    - UDP services both inbound and outbound
    - ICMP packet types both inbound and outbound
    - Blocking ports inbound and outbound regardless of IP or protocol
    - Set Maximum Throughput and Minimum Delay flags for user defined protocols
- Only allow **stateful** filtering via NEW and ESTABLISHED traffic to go through the firewall.
- Must ensure that the firewall rejects inbound connection requests unless it’s to a permitted service.

Design

The following diagram shows how the network architecture is set up:



Note that the “Standalone Firewall” is actually one of the machines in the lab, including the internal host but with routing configuration via **route** command, connecting Ethernet cables, and changing network card settings, they will act as if it were a separate subnet.

The “Other Firewall” is a firewall for the lab machines that is blocked off from the main BCIT network.

Test Case Table

No.	Test Case	Tools Used	Expected Outcome	Pass/Fail
1	Inbound/Outbound TCP packets allowed on user defined ports	hping3	Only certain TCP packets should be allowed through.	PASS. Details are attached below.
2	Inbound/Outbound UDP packets allowed on user defined ports	hping3	Only certain UDP packets should be allowed through.	PASS. Details are attached below.
3	Inbound/Outbound ICMP packets allowed on user defined icmp types	hping3	Only certain ICMP packets should be allowed through.	PASS. Details are attached below.
4	All packets are dropped destined to the firewall host from outside	hping3	Everything except DNS and DHCP should be dropped.	PASS. Details are attached below.
5	Drop SYN packets to high ports	hping3	New Connections to high ports should be dropped.	PASS. Details are attached below.
6	Drop SYN packets from low ports	hping3	New connections from low ports should be dropped.	PASS. Details are attached below.
7	Accept Fragments	hping3	Fragmented packets should be allowed through.	PASS. Details are attached below.
8	Accept packets on existing connections	hping3	Packets on existing connections should be allowed through.	PASS. Details are attached below.
9	Drop external packets with source address matching internal network	hping3	External packets with src address matching	PASS. Details are attached below.

			internal network should be dropped.	
10	Drop all tcp packets with SYN and FIN flags set	hping3	TCP packets should be dropped with SYN and FIN flags set.	PASS. Details are attached below.
11	Drop all Telnet packets via from port 23	hping3	Telnet packets should be dropped from port 23.	PASS. Details are attached below.
12	Drop all UDP packets to ports 32768 – 32775, 137 – 139	hping3	UDP packets should be dropped from defined ports.	PASS. Details are attached below.
13	Drop all TCP packets to ports 32768 – 32775, 137 – 139, 111 and 515	hping3	TCP packets should be dropped from defined ports.	PASS. Details are attached below.
14	Mangle FTP and SSH to Minimum Delay	hping3, SSH command, some FTP command	FTP and SSH packets should get mangled for Minimum Delay.	PASS. Details are attached below.
15	Mangle FTP data to Maximum Throughput	hping3, some FTP command	FTP data packets should be mangled for Maximum Throughput.	PASS. Details are attached below.
16	Drop as Default Policy	hping3	All other packets outside of defined rules should be dropped.	PASS. Details are attached below.

Test Cases:

On an external host, we ran the testing script ‘testext.sh’ and on the internal host, we ran the testing script ‘testint.sh’. The output for each test case went to their respective file in folders internal_tests/ and external_tests/. This output has been supplemented with screenshots of Wireshark being run on relevant network interfaces.

Test Case 1

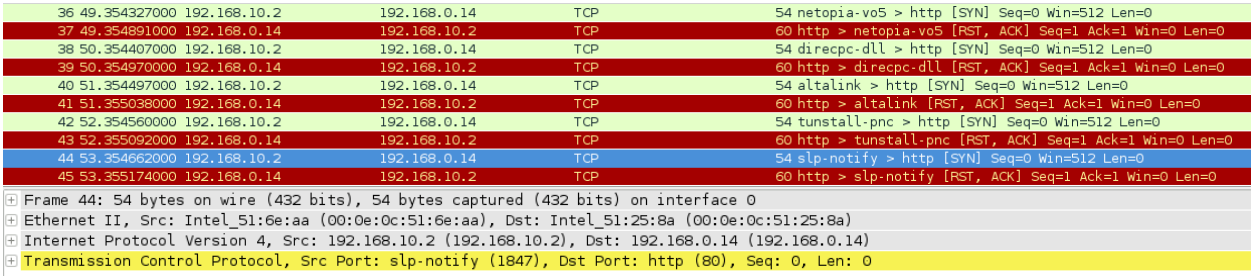
This test case was to ensure that inbound and outbound TCP connections on user defined ports are able to be established. We are testing on ports 22, 80, and 443 for outbound connections from an internal machine, and on port 22, 80 and 443 for inbound connections from an external machine.

Internal Test

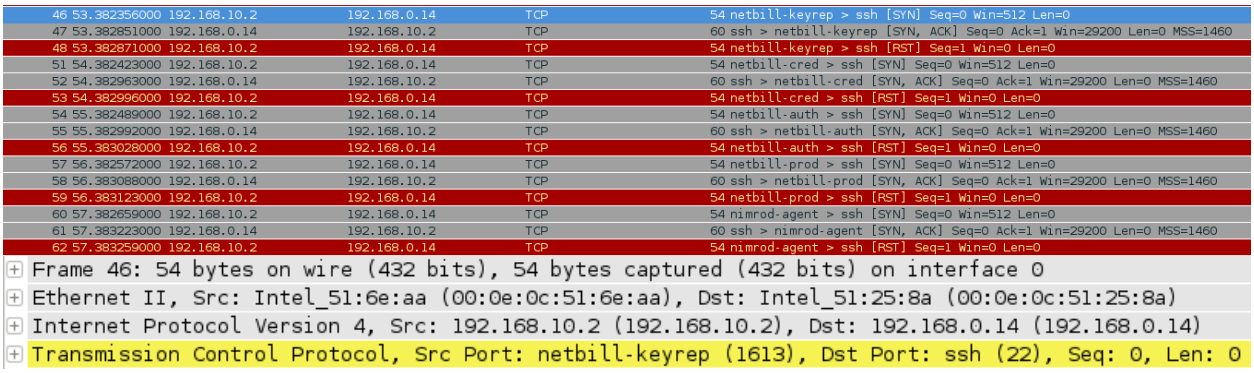
Screenshot of Test Script being run:

```
Test case 1 commencing...
Pinging 5 TCP packets to port 80 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.6/0.6/0.6 ms
Pinging 5 TCP packets to port 22 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.5/0.6/0.6 ms
Pinging 5 TCP packets to port 443 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.5/0.6/0.6 ms
SSH Login of root@192.168.0.14
Test case 1 results written to file
```

Wireshark screenshot for http traffic on port 80:

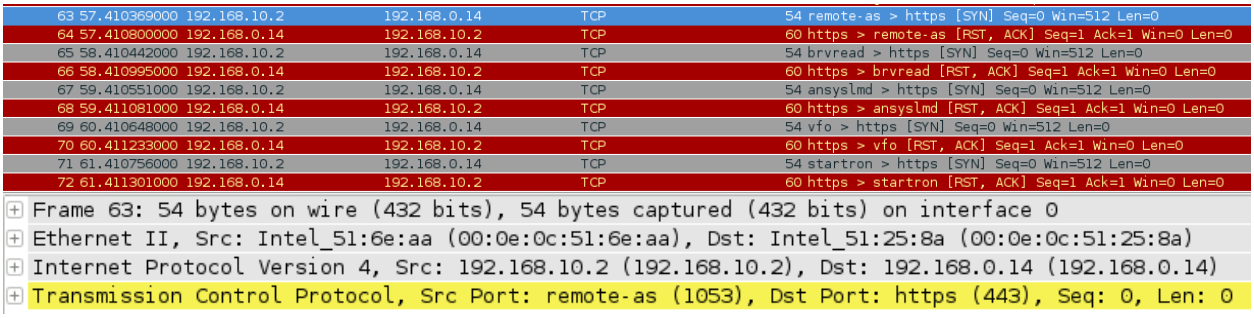


Wireshark screenshot for ssh traffic on port 22:



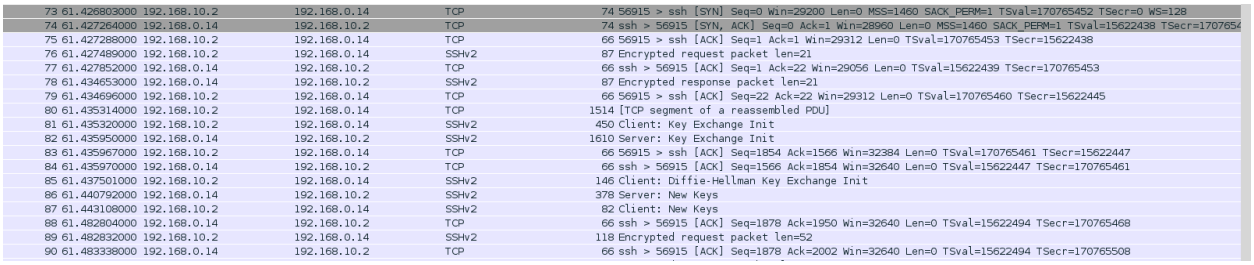
46	53.382356000	192.168.10.2	192.168.0.14	TCP	54 netbill-keyrep > ssh [SYN] Seq=0 Win=512 Len=0
47	53.382851000	192.168.0.14	192.168.10.2	TCP	60 ssh > netbill-keyrep [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
48	53.382871000	192.168.10.2	192.168.0.14	TCP	54 netbill-keyrep > ssh [RST] Seq=1 Win=0 Len=0
51	54.382423000	192.168.10.2	192.168.0.14	TCP	54 netbill-cred > ssh [SYN] Seq=0 Win=512 Len=0
52	54.382963000	192.168.0.14	192.168.10.2	TCP	60 ssh > netbill-cred [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
53	54.382996000	192.168.10.2	192.168.0.14	TCP	54 netbill-cred > ssh [RST] Seq=1 Win=0 Len=0
54	55.382489000	192.168.10.2	192.168.0.14	TCP	54 netbill-auth > ssh [SYN] Seq=0 Win=512 Len=0
55	55.382992000	192.168.0.14	192.168.10.2	TCP	60 ssh > netbill-auth [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
56	55.383028000	192.168.10.2	192.168.0.14	TCP	54 netbill-auth > ssh [RST] Seq=1 Win=0 Len=0
57	56.382572000	192.168.10.2	192.168.0.14	TCP	54 netbill-prod > ssh [SYN] Seq=0 Win=512 Len=0
58	56.383088000	192.168.0.14	192.168.10.2	TCP	60 ssh > netbill-prod [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
59	56.383123000	192.168.10.2	192.168.0.14	TCP	54 netbill-prod > ssh [RST] Seq=1 Win=0 Len=0
60	57.382659000	192.168.10.2	192.168.0.14	TCP	54 nimrod-agent > ssh [SYN] Seq=0 Win=512 Len=0
61	57.383223000	192.168.0.14	192.168.10.2	TCP	60 ssh > nimrod-agent [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
62	57.383259000	192.168.10.2	192.168.0.14	TCP	54 nimrod-agent > ssh [RST] Seq=1 Win=0 Len=0

Wireshark screenshot for https on port 443:



63	57.410369000	192.168.10.2	192.168.0.14	TCP	54 remote-as > https [SYN] Seq=0 Win=512 Len=0
64	57.410800000	192.168.0.14	192.168.10.2	TCP	60 https > remote-as [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
65	58.410442000	192.168.10.2	192.168.0.14	TCP	54 brvread > https [SYN] Seq=0 Win=512 Len=0
66	58.410995000	192.168.0.14	192.168.10.2	TCP	60 https > brvread [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
67	59.410551000	192.168.10.2	192.168.0.14	TCP	54 ansyslmd > https [SYN] Seq=0 Win=512 Len=0
68	59.411081000	192.168.0.14	192.168.10.2	TCP	60 https > ansyslmd [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
69	60.410648000	192.168.10.2	192.168.0.14	TCP	54 vfo > https [SYN] Seq=0 Win=512 Len=0
70	60.411233000	192.168.0.14	192.168.10.2	TCP	60 https > vfo [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
71	61.410756000	192.168.10.2	192.168.0.14	TCP	54 startron > https [SYN] Seq=0 Win=512 Len=0
72	61.411301000	192.168.0.14	192.168.10.2	TCP	60 https > startron [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Ssh Connection in Wireshark. This shows that a full connection can be made and torn down:



73	61.426803000	192.168.10.2	192.168.0.14	TCP	74 56915 > ssh [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=170765452 TSecr=0 WS=128
74	61.427264000	192.168.0.14	192.168.10.2	TCP	74 ssh > 56915 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=15622438 TSecr=1707654
75	61.427288000	192.168.10.2	192.168.0.14	TCP	66 56915 > ssh [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=170765453 TSecr=15622438
76	61.427489000	192.168.10.2	192.168.0.14	SSHv2	87 Encrypted request packet len=21
77	61.427852000	192.168.0.14	192.168.10.2	TCP	66 ssh > 56915 [ACK] Seq=1 Ack=22 Win=29056 Len=0 TSval=15622439 TSecr=170765453
78	61.434653000	192.168.0.14	192.168.10.2	SSHv2	87 Encrypted response packet len=21
79	61.434696000	192.168.10.2	192.168.0.14	TCP	66 56915 > ssh [ACK] Seq=22 Ack=22 Win=29312 Len=0 TSval=170765460 TSecr=15622445
80	61.435314000	192.168.10.2	192.168.0.14	TCP	1514 [TCP segment of a reassembled PDU]
81	61.435320000	192.168.10.2	192.168.0.14	SSHv2	450 Client: Key Exchange Init
82	61.435950000	192.168.0.14	192.168.10.2	SSHv2	1610 Server: Key Exchange Init
83	61.435967000	192.168.10.2	192.168.0.14	TCP	66 56915 > ssh [ACK] Seq=1854 Ack=1566 Win=32384 Len=0 TSval=170765461 TSecr=15622447
84	61.435970000	192.168.0.14	192.168.10.2	TCP	66 ssh > 56915 [ACK] Seq=1566 Ack=1854 Win=32640 Len=0 TSval=15622447 TSecr=170765461
85	61.437501000	192.168.10.2	192.168.0.14	SSHv2	146 Client: Diffie-Hellman Key Exchange Init
86	61.440792000	192.168.0.14	192.168.10.2	SSHv2	378 Server: New Keys
87	61.443108000	192.168.10.2	192.168.0.14	SSHv2	82 Client: New Keys
88	61.482804000	192.168.0.14	192.168.10.2	TCP	66 ssh > 56915 [ACK] Seq=1878 Ack=1950 Win=32640 Len=0 TSval=15622494 TSecr=170765468
89	61.482832000	192.168.10.2	192.168.0.14	SSHv2	118 Encrypted request packet len=52
90	61.483338000	192.168.0.14	192.168.10.2	TCP	66 ssh > 56915 [ACK] Seq=1878 Ack=2002 Win=32640 Len=0 TSval=15622494 TSecr=170765508

Test Output:

test1

```
HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.14 ttl=63 DF id=10335 sport=80 flags=RA seq=0 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10336 sport=80 flags=RA seq=1 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10337 sport=80 flags=RA seq=2 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10338 sport=80 flags=RA seq=3 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10339 sport=80 flags=RA seq=4 win=0 rtt=0.6 ms
HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.14 ttl=63 DF id=0 sport=22 flags=SA seq=0 win=29200 rtt=0.5 ms
len=46 ip=192.168.0.14 ttl=63 DF id=0 sport=22 flags=SA seq=1 win=29200 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=0 sport=22 flags=SA seq=2 win=29200 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=0 sport=22 flags=SA seq=3 win=29200 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=0 sport=22 flags=SA seq=4 win=29200 rtt=0.6 ms
HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.14 ttl=63 DF id=10340 sport=443 flags=RA seq=0 win=0 rtt=0.5 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10341 sport=443 flags=RA seq=1 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10342 sport=443 flags=RA seq=2 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10343 sport=443 flags=RA seq=3 win=0 rtt=0.6 ms
len=46 ip=192.168.0.14 ttl=63 DF id=10344 sport=443 flags=RA seq=4 win=0 rtt=0.6 ms
em1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.14  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::7a2b:cbff:fea3:d737  prefixlen 64  scopeid 0x20<link>
        ether 78:2b:cb:a3:d7:37  txqueuelen 1000  (Ethernet)
        RX packets 58024  bytes 33337658 (31.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 62489  bytes 9028296 (8.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 20  memory 0xe1b00000-e1b20000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 504  bytes 39560 (38.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 504  bytes 39560 (38.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

```

From the above screenshots, notice that the connections to the outside host were successful. The ‘ifconfig’ tool was run during the ssh connection to show that it was connecting to the external host and not the stand-alone firewall.

External Test

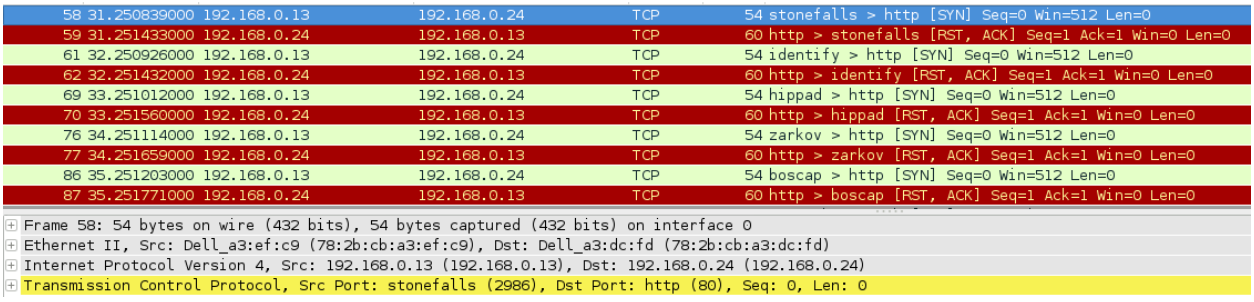
Excerpt from test script:

```
10 ################################# Test Case 1 ###################################
11 echo "Test case 1 commencing..."
12 CASE=1
13 # Allow Inbound/Outbound TCP Packets allowed on user defined port 80, 22, 443
14
15 echo "Pinging 5 TCP packets to port 80 of host $IP:"
16 hping3 $IP -S -c 5 -p 80 > $BASEFILE$CASE
17
18 echo "Pinging 5 TCP packets to port 22 of host $IP:"
19 hping3 $IP -S -c 5 -p 22 >> $BASEFILE$CASE
20
21 echo "Pinging 5 TCP packets to port 443 of host $IP:"
22 hping3 $IP -S -c 5 -p 443 >> $BASEFILE$CASE
23
24 echo "SSH Login of $SSH_ADDR"
25 sshpass -p "uest1onQ?" ssh -o StrictHostKeyChecking=no $SSH_ADDR "ifconfig;exit" >> $BASEFILE$CASE
26
27 echo "Test case 1 results written to file"
```

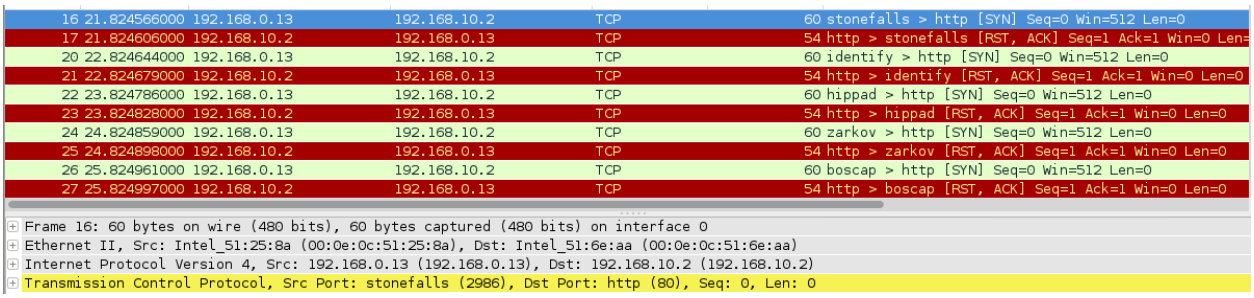
Screenshot of test script being run:

```
Test case 1 commencing...
Pinging 5 TCP packets to port 80 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.6/0.6/0.7 ms
Pinging 5 TCP packets to port 22 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.6/0.6/0.7 ms
Pinging 5 TCP packets to port 443 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.5/0.6/0.7 ms
SSH Login of root@192.168.0.24
Test case 1 results written to file
```

Screenshot of Wireshark for http on external host:



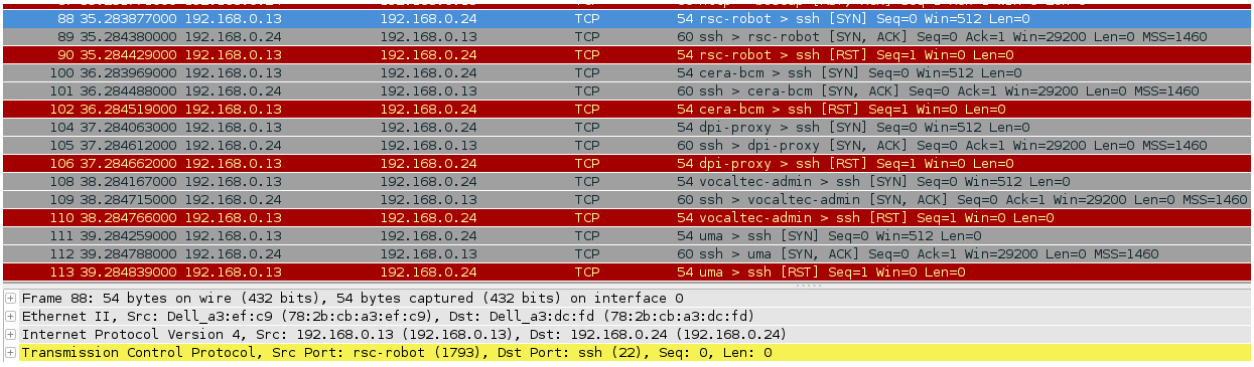
Screenshot of Wireshark for http on internal host:



16	21.824566000	192.168.0.13	192.168.10.2	TCP	60 stonefalls > http [SYN] Seq=0 Win=512 Len=0
17	21.824606000	192.168.10.2	192.168.0.13	TCP	54 http > stonefalls [RST, ACK] Seq=1 Ack=1 Win=0 Len=
20	22.824644000	192.168.0.13	192.168.10.2	TCP	60 identify > http [SYN] Seq=0 Win=512 Len=0
21	22.824679000	192.168.10.2	192.168.0.13	TCP	54 http > identify [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
22	23.824786000	192.168.0.13	192.168.10.2	TCP	60 hippad > http [SYN] Seq=0 Win=512 Len=0
23	23.824828000	192.168.10.2	192.168.0.13	TCP	54 http > hippad [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
24	24.824859000	192.168.0.13	192.168.10.2	TCP	60 zarkov > http [SYN] Seq=0 Win=512 Len=0
25	24.824898000	192.168.10.2	192.168.0.13	TCP	54 http > zarkov [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
26	25.824961000	192.168.0.13	192.168.10.2	TCP	60 boscap > http [SYN] Seq=0 Win=512 Len=0
27	25.824997000	192.168.10.2	192.168.0.13	TCP	54 http > boscap [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Frame 16: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Intel_51:25:8a (00:0e:0c:51:25:8a), Dst: Intel_51:6e:aa (00:0e:0c:51:6e:aa)
Internet Protocol Version 4, Src: 192.168.0.13 (192.168.0.13), Dst: 192.168.10.2 (192.168.10.2)
Transmission Control Protocol, Src Port: stonefalls (2986), Dst Port: http (80), Seq: 0, Len: 0

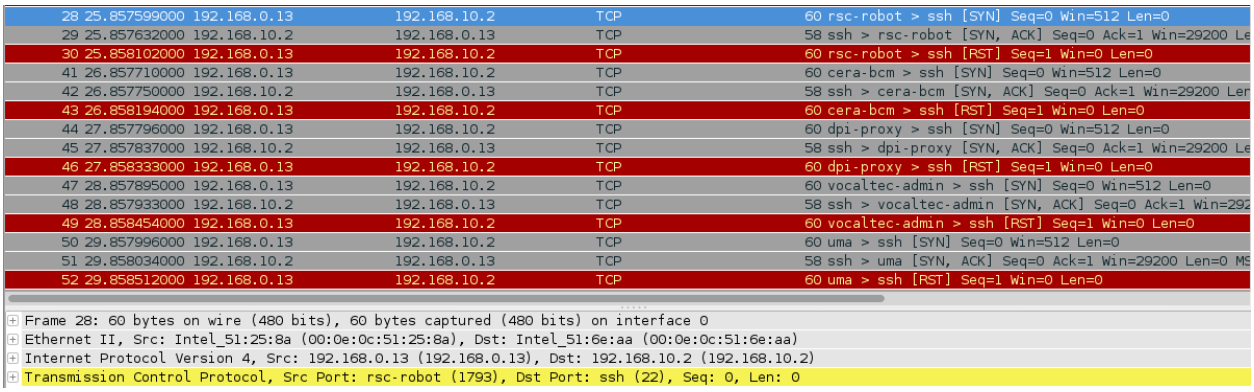
Screenshot of Wireshark for ssh on external host:



88	35.283877000	192.168.0.13	192.168.0.24	TCP	54 rsc-robot > ssh [SYN] Seq=0 Win=512 Len=0
89	35.284380000	192.168.0.24	192.168.0.13	TCP	60 ssh > rsc-robot [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
90	35.284429000	192.168.0.13	192.168.0.24	TCP	54 rsc-robot > ssh [RST] Seq=1 Win=0 Len=0
100	36.283969000	192.168.0.13	192.168.0.24	TCP	54 cera-bcm > ssh [SYN] Seq=0 Win=512 Len=0
101	36.284488000	192.168.0.24	192.168.0.13	TCP	60 ssh > cera-bcm [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
102	36.284519000	192.168.0.13	192.168.0.24	TCP	54 cera-bcm > ssh [RST] Seq=1 Win=0 Len=0
104	37.284063000	192.168.0.13	192.168.0.24	TCP	54 dpi-proxy > ssh [SYN] Seq=0 Win=512 Len=0
105	37.284612000	192.168.0.24	192.168.0.13	TCP	60 ssh > dpi-proxy [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
106	37.284662000	192.168.0.13	192.168.0.24	TCP	54 dpi-proxy > ssh [RST] Seq=1 Win=0 Len=0
108	38.284167000	192.168.0.13	192.168.0.24	TCP	54 vocaltec-admin > ssh [SYN] Seq=0 Win=512 Len=0
109	38.284715000	192.168.0.24	192.168.0.13	TCP	60 ssh > vocaltec-admin [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
110	38.284766000	192.168.0.13	192.168.0.24	TCP	54 vocaltec-admin > ssh [RST] Seq=1 Win=0 Len=0
111	39.284259000	192.168.0.13	192.168.0.24	TCP	54 uma > ssh [SYN] Seq=0 Win=512 Len=0
112	39.284788000	192.168.0.24	192.168.0.13	TCP	60 ssh > uma [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
113	39.284839000	192.168.0.13	192.168.0.24	TCP	54 uma > ssh [RST] Seq=1 Win=0 Len=0

Frame 88: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Dell_a3:ef:c9 (78:2b:cb:a3:ef:c9), Dst: Dell_a3:dc:fd (78:2b:cb:a3:dc:fd)
Internet Protocol Version 4, Src: 192.168.0.13 (192.168.0.13), Dst: 192.168.0.24 (192.168.0.24)
Transmission Control Protocol, Src Port: rsc-robot (1793), Dst Port: ssh (22), Seq: 0, Len: 0

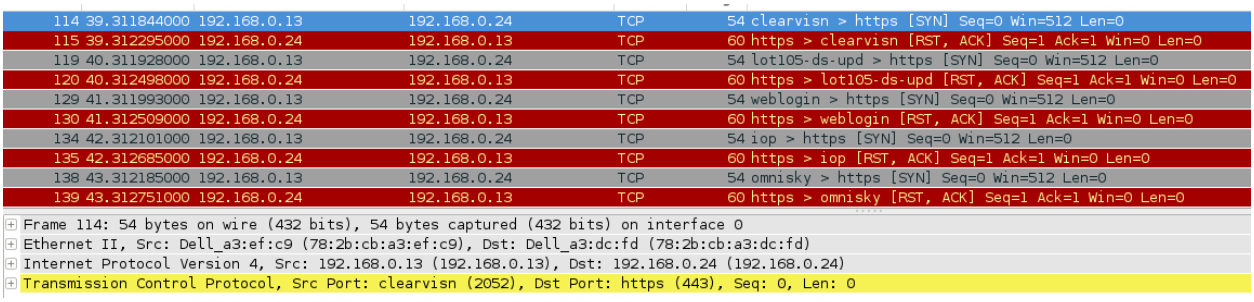
Screenshot of Wireshark for ssh on internal host:



28	25.857599000	192.168.0.13	192.168.10.2	TCP	60 rsc-robot > ssh [SYN] Seq=0 Win=512 Len=0
29	25.857632000	192.168.10.2	192.168.0.13	TCP	58 ssh > rsc-robot [SYN, ACK] Seq=0 Ack=1 Win=29200 Le
30	25.858102000	192.168.0.13	192.168.10.2	TCP	60 rsc-robot > ssh [RST] Seq=1 Win=0 Len=0
41	26.857710000	192.168.0.13	192.168.10.2	TCP	60 cera-bcm > ssh [SYN] Seq=0 Win=512 Len=0
42	26.857750000	192.168.10.2	192.168.0.13	TCP	58 ssh > cera-bcm [SYN, ACK] Seq=0 Ack=1 Win=29200 Len
43	26.858194000	192.168.0.13	192.168.10.2	TCP	60 cera-bcm > ssh [RST] Seq=1 Win=0 Len=0
44	27.857796000	192.168.0.13	192.168.10.2	TCP	60 dpi-proxy > ssh [SYN] Seq=0 Win=512 Len=0
45	27.857837000	192.168.10.2	192.168.0.13	TCP	58 ssh > dpi-proxy [SYN, ACK] Seq=0 Ack=1 Win=29200 Le
46	27.858333000	192.168.0.13	192.168.10.2	TCP	60 dpi-proxy > ssh [RST] Seq=1 Win=0 Len=0
47	28.857895000	192.168.0.13	192.168.10.2	TCP	60 vocaltec-admin > ssh [SYN] Seq=0 Win=512 Len=0
48	28.857933000	192.168.10.2	192.168.0.13	TCP	58 ssh > vocaltec-admin [SYN, ACK] Seq=0 Ack=1 Win=292
49	28.858454000	192.168.0.13	192.168.10.2	TCP	60 vocaltec-admin > ssh [RST] Seq=1 Win=0 Len=0
50	29.857996000	192.168.0.13	192.168.10.2	TCP	60 uma > ssh [SYN] Seq=0 Win=512 Len=0
51	29.858034000	192.168.10.2	192.168.0.13	TCP	58 ssh > uma [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS
52	29.858512000	192.168.0.13	192.168.10.2	TCP	60 uma > ssh [RST] Seq=1 Win=0 Len=0

Frame 28: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Intel_51:25:8a (00:0e:0c:51:25:8a), Dst: Intel_51:6e:aa (00:0e:0c:51:6e:aa)
Internet Protocol Version 4, Src: 192.168.0.13 (192.168.0.13), Dst: 192.168.10.2 (192.168.10.2)
Transmission Control Protocol, Src Port: rsc-robot (1793), Dst Port: ssh (22), Seq: 0, Len: 0

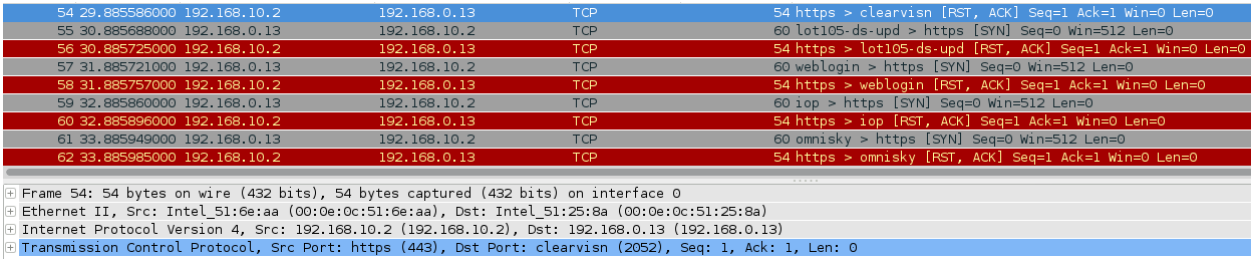
Screenshot of Wireshark for https on external host:



114	39.311844000	192.168.0.13	192.168.0.24	TCP	54 clearvisn > https [SYN] Seq=0 Win=512 Len=0
115	39.312295000	192.168.0.24	192.168.0.13	TCP	60 https > clearvisn [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
119	40.311928000	192.168.0.13	192.168.0.24	TCP	54 lot105-ds-upd > https [SYN] Seq=0 Win=512 Len=0
120	40.312498000	192.168.0.24	192.168.0.13	TCP	60 https > lot105-ds-upd [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
129	41.311993000	192.168.0.13	192.168.0.24	TCP	54 weblogin > https [SYN] Seq=0 Win=512 Len=0
130	41.312509000	192.168.0.24	192.168.0.13	TCP	60 https > weblogin [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
134	42.312101000	192.168.0.13	192.168.0.24	TCP	54 iop > https [SYN] Seq=0 Win=512 Len=0
135	42.312685000	192.168.0.24	192.168.0.13	TCP	60 https > iop [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
138	43.312185000	192.168.0.13	192.168.0.24	TCP	54 omnisky > https [SYN] Seq=0 Win=512 Len=0
139	43.312751000	192.168.0.24	192.168.0.13	TCP	60 https > omnisky [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Frame 114: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Dell_a3:ef:c9 (78:2b:cb:a3:ef:c9), Dst: Dell_a3:dc:fd (78:2b:cb:a3:dc:fd)
Internet Protocol Version 4, Src: 192.168.0.13 (192.168.0.13), Dst: 192.168.0.24 (192.168.0.24)
Transmission Control Protocol, Src Port: clearvisn (2052), Dst Port: https (443), Seq: 0, Len: 0

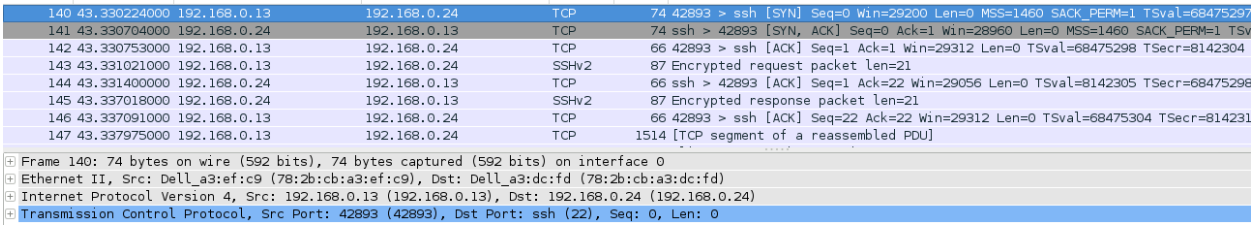
Screenshot of Wireshark for https on internal host:



54	29.885586000	192.168.10.2	192.168.0.13	TCP	54 https > clearvisn [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
55	30.885688000	192.168.0.13	192.168.10.2	TCP	60 lot105-ds-upd > https [SYN] Seq=0 Win=512 Len=0
56	30.885725000	192.168.10.2	192.168.0.13	TCP	54 https > lot105-ds-upd [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
57	31.885721000	192.168.0.13	192.168.10.2	TCP	60 weblogin > https [SYN] Seq=0 Win=512 Len=0
58	31.885757000	192.168.10.2	192.168.0.13	TCP	54 https > weblogin [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
59	32.885860000	192.168.0.13	192.168.10.2	TCP	60 iop > https [SYN] Seq=0 Win=512 Len=0
60	32.885896000	192.168.10.2	192.168.0.13	TCP	54 https > iop [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
61	33.885949000	192.168.0.13	192.168.10.2	TCP	60 omnisky > https [SYN] Seq=0 Win=512 Len=0
62	33.885985000	192.168.10.2	192.168.0.13	TCP	54 https > omnisky [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Frame 54: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Intel_51:6e:aa (00:0e:0c:51:6e:aa), Dst: Intel_51:25:8a (00:0e:0c:51:25:8a)
Internet Protocol Version 4, Src: 192.168.10.2 (192.168.10.2), Dst: 192.168.0.13 (192.168.0.13)
Transmission Control Protocol, Src Port: https (443), Dst Port: clearvisn (2052), Seq: 1, Ack: 1, Len: 0

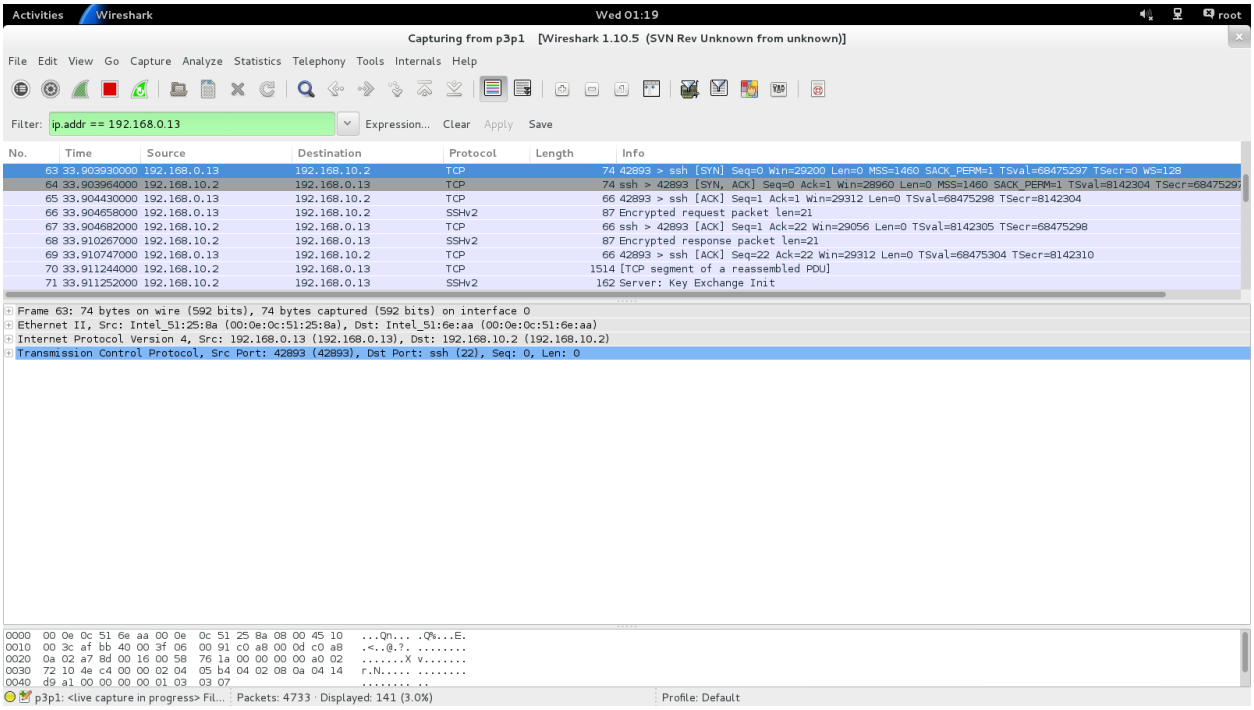
Screenshot of Wireshark for ssh connection on external host:



140	43.330224000	192.168.0.13	192.168.0.24	TCP	74 42893 > ssh [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=68475297
141	43.330704000	192.168.0.24	192.168.0.13	TCP	74 ssh > 42893 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSv
142	43.330753000	192.168.0.13	192.168.0.24	TCP	66 42893 > ssh [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=68475298 TSecr=8142304
143	43.331021000	192.168.0.13	192.168.0.24	SSHv2	87 Encrypted request packet len=21
144	43.331400000	192.168.0.24	192.168.0.13	TCP	66 ssh > 42893 [ACK] Seq=1 Ack=22 Win=29056 Len=0 TSval=8142305 TSecr=68475298
145	43.337018000	192.168.0.24	192.168.0.13	SSHv2	87 Encrypted response packet len=21
146	43.337091000	192.168.0.13	192.168.0.24	TCP	66 42893 > ssh [ACK] Seq=22 Ack=22 Win=29312 Len=0 TSval=68475304 TSecr=814231
147	43.337975000	192.168.0.13	192.168.0.24	TCP	1514 [TCP segment of a reassembled PDU]

Frame 140: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
Ethernet II, Src: Dell_a3:ef:c9 (78:2b:cb:a3:ef:c9), Dst: Dell_a3:dc:fd (78:2b:cb:a3:dc:fd)
Internet Protocol Version 4, Src: 192.168.0.13 (192.168.0.13), Dst: 192.168.0.24 (192.168.0.24)
Transmission Control Protocol, Src Port: 42893 (42893), Dst Port: ssh (22), Seq: 0, Len: 0

Screenshot of Wireshark for ssh connection on internal host:



Test Output:

test1

```
HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.24 ttl=63 DF id=34265 sport=80 flags=RA seq=0 win=0 rtt=0.7 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34266 sport=80 flags=RA seq=1 win=0 rtt=0.6 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34267 sport=80 flags=RA seq=2 win=0 rtt=0.6 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34268 sport=80 flags=RA seq=3 win=0 rtt=0.6 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34269 sport=80 flags=RA seq=4 win=0 rtt=0.7 ms
HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=0 win=29200 rtt=0.6 ms
len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=1 win=29200 rtt=0.6 ms
len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=2 win=29200 rtt=0.7 ms
len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=3 win=29200 rtt=0.7 ms
len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=4 win=29200 rtt=0.6 ms
HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.24 ttl=63 DF id=34270 sport=443 flags=RA seq=0 win=0 rtt=0.5 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34271 sport=443 flags=RA seq=1 win=0 rtt=0.7 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34272 sport=443 flags=RA seq=2 win=0 rtt=0.6 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34273 sport=443 flags=RA seq=3 win=0 rtt=0.7 ms
len=46 ip=192.168.0.24 ttl=63 DF id=34274 sport=443 flags=RA seq=4 win=0 rtt=0.7 ms
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 8135  bytes 709387 (692.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8135  bytes 709387 (692.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

p3p1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.10.2  netmask 255.255.255.0  broadcast 192.168.10.255
        inet6 fe80::20e:cff:fe51:6eaa  prefixlen 64  scopeid 0x20<link>
        ether 00:0e:0c:51:6e:aa  txqueuelen 1000  (Ethernet)
        RX packets 14949  bytes 11921050 (11.3 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 22426  bytes 3137815 (2.9 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

```

From the output, notice that the connections to the internal host were successful. The ‘ifconfig’ tool was run as well during this ssh connection to show that it was connecting to the internal host and not the stand-alone firewall. From the Wireshark screenshots we see that the Port Forwarding, DNAT, worked correctly.

Test Case 2

The following test was run to establish a UDP connection on user defined protocols. For this test, we chose UDP port 80 even though the port will not be open on the receiving host. This means that there will be no response back.

Internal Test

Excerpt from test script:

```
28 ################################## Test Case 2 ###################################
29 echo "Test case 2 commencing..."
30 CASE=2
31 # Allow Inbound/Outbound UDP Packets allowed on user defined port 80
32
33 echo "Pinging 5 UDP Packets to port 80 of host $IP:"
34 hping3 $IP --udp -c 5 -p 80 > $BASEFILE$CASE
35
36 echo "Test case 2 results written to file"
```

Screenshot of test script being run:

```
Test case 2 commencing...
Pinging 5 UDP Packets to port 80 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 2 results written to file
```

Output of test:

test2

```
1 HPING 192.168.0.14 (p3p1 192.168.0.14): udp mode set, 28 headers + 0 data bytes
```

Wireshark Screenshot on internal host:

109	61.524337000	192.168.10.2	192.168.0.14	UDP	42 Source port: iqserver Destination port: http
110	62.524428000	192.168.10.2	192.168.0.14	UDP	42 Source port: ncr-ccl Destination port: http
111	63.524490000	192.168.10.2	192.168.0.14	UDP	42 Source port: utsftp Destination port: http
112	64.524563000	192.168.10.2	192.168.0.14	UDP	42 Source port: vrcommerce Destination port: http
113	65.524621000	192.168.10.2	192.168.0.14	UDP	42 Source port: ito-e-gui Destination port: http

Wireshark screenshot on external host:

141	81.308243000	192.168.0.24	192.168.0.14	UDP	60 Source port: iqserver Destination port: http
142	81.308274000	192.168.0.14	192.168.0.24	ICMP	70 Destination unreachable (Port unreachable)
144	82.308371000	192.168.0.24	192.168.0.14	UDP	60 Source port: ncr-ccl Destination port: http
145	82.308412000	192.168.0.14	192.168.0.24	ICMP	70 Destination unreachable (Port unreachable)
146	83.308450000	192.168.0.24	192.168.0.14	UDP	60 Source port: utsftp Destination port: http
147	83.308490000	192.168.0.14	192.168.0.24	ICMP	70 Destination unreachable (Port unreachable)
149	84.308517000	192.168.0.24	192.168.0.14	UDP	60 Source port: vrcommerce Destination port: http
150	84.308556000	192.168.0.14	192.168.0.24	ICMP	70 Destination unreachable (Port unreachable)
153	85.308575000	192.168.0.24	192.168.0.14	UDP	60 Source port: ito-e-gui Destination port: http
154	85.308612000	192.168.0.14	192.168.0.24	ICMP	70 Destination unreachable (Port unreachable)

From this test, even though there was no response to the UDP packets, we know that the udp packets were passed through due to the ICMP messages(Port unreachable) generated by the external host.

External Test

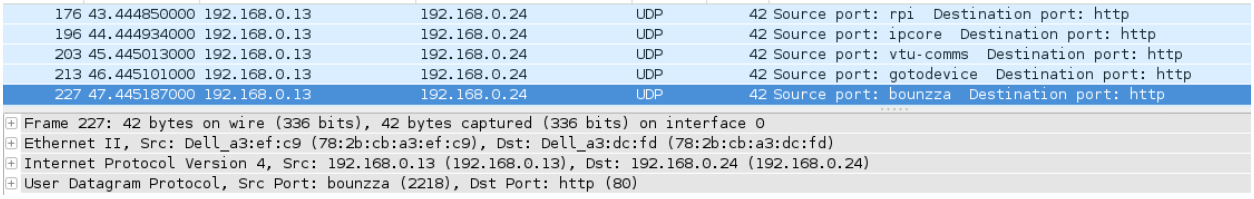
Excerpt from test script:

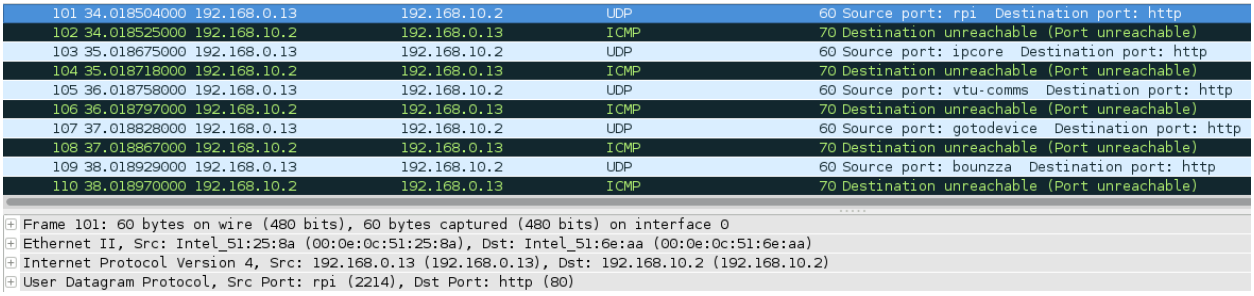
```
29 ################################## Test Case 2 ###################################
30 echo "Test case 2 commencing..."
31 CASE=2
32 # Allow Inbound/Outbound UDP Packets allowed on user defined port 80
33
34 echo "Pinging 5 UDP Packets to port 80 of host $IP:"
35 hping3 $IP --udp -c 5 -p 80 > $BASEFILE$CASE
36
37 echo "Test case 2 results written to file"
38
39
```

Screenshot of test script being run:

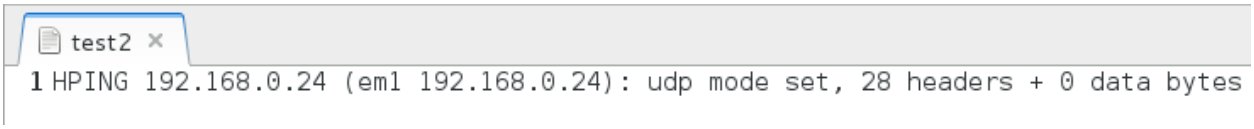
```
Test case 2 commencing...
Pinging 5 UDP Packets to port 80 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 2 results written to file
```

Screenshot of Wireshark:





Test Output:



From this test, even though there was no response to the UDP packets, we know that the udp packets were passed through due to the ICMP messages(Port unreachable) generated by the internal host. Also, from the Wireshark screenshots we see that the Port Forwarding, DNAT, worked correctly on user defined ports.

Test Case 3

The following test was run to establish a ICMP connection on user defined ICMP types. For this test we used ICMP types 8 and 0 for ping and ping reply. Refer to test case 2 for ICMP packets that did not make it through the firewall.

Internal Test

Excerpt from test script:

```
################################ Test Case 3 ###################################
echo "Test case 3 commencing..."
CASE=3
# Allow Inbound/Outbound ICMP Packets allowed on type 8

echo "Pinging 5 ICMP packets to host $IP:"
hping3 $IP --icmp -C 8 -c 5 > $BASEFILE$CASE
echo "Test case 3 results written to file"

```

Screenshot of test script running:

```
Test case 3 commencing...
Pinging 5 ICMP packets to host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.5/0.5/0.5 ms
Test case 3 results written to file
```

Wireshark screenshot of ping requests and replies on internal host:

114	66.552347000	192.168.10.2	192.168.0.14	ICMP	42 Echo (ping) request id=0xcd27, seq=0/0, ttl=64 (reply in 115)
115	66.552816000	192.168.0.14	192.168.10.2	ICMP	60 Echo (ping) reply id=0xcd27, seq=0/0, ttl=63 (request in 114)
116	67.552423000	192.168.10.2	192.168.0.14	ICMP	42 Echo (ping) request id=0xcd27, seq=256/1, ttl=64 (reply in 117)
117	67.552901000	192.168.0.14	192.168.10.2	ICMP	60 Echo (ping) reply id=0xcd27, seq=256/1, ttl=63 (request in 116)
118	68.552495000	192.168.10.2	192.168.0.14	ICMP	42 Echo (ping) request id=0xcd27, seq=512/2, ttl=64 (reply in 119)
119	68.552976000	192.168.0.14	192.168.10.2	ICMP	60 Echo (ping) reply id=0xcd27, seq=512/2, ttl=63 (request in 118)
120	69.552584000	192.168.10.2	192.168.0.14	ICMP	42 Echo (ping) request id=0xcd27, seq=768/3, ttl=64 (reply in 121)
121	69.553073000	192.168.0.14	192.168.10.2	ICMP	60 Echo (ping) reply id=0xcd27, seq=768/3, ttl=63 (request in 120)
122	70.552668000	192.168.10.2	192.168.0.14	ICMP	42 Echo (ping) request id=0xcd27, seq=1024/4, ttl=64 (reply in 123)
123	70.553152000	192.168.0.14	192.168.10.2	ICMP	60 Echo (ping) reply id=0xcd27, seq=1024/4, ttl=63 (request in 122)

Test Output:

test3

```
HPING 192.168.0.14 (p3p1 192.168.0.14): icmp mode set, 28 headers + 0 data bytes
len=46 ip=192.168.0.14 ttl=63 id=10350 icmp_seq=0 rtt=0.5 ms
len=46 ip=192.168.0.14 ttl=63 id=10351 icmp_seq=1 rtt=0.5 ms
len=46 ip=192.168.0.14 ttl=63 id=10352 icmp_seq=2 rtt=0.5 ms
len=46 ip=192.168.0.14 ttl=63 id=10353 icmp_seq=3 rtt=0.5 ms
len=46 ip=192.168.0.14 ttl=63 id=10354 icmp_seq=4 rtt=0.5 ms
```

Screenshot of iptables -L -n -x -v:

```
Chain icmpin (1 references)
    pkts      bytes target     prot opt in     out     source               destination
       5      140 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0           i
cmptype 0 state NEW,ESTABLISHED
       5      140 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0           i
cmptype 8 state NEW,ESTABLISHED
```

From the above screenshots, notice that the icmp packets and replies made it through the firewall.

External Test

Excerpt from test script:

```
40 ################################# Test Case 3 ###################################
41 echo "Test case 3 commencing..."
42 CASE=3
43 # Allow Inbound/Outbound ICMP Packets allowed on type 8
44
45 echo "Pinging 5 ICMP packets to host $IP:"
46 hping3 $IP --icmp -C 8 -c 5 > $BASEFILE$CASE
47 echo "Test case 3 results written to file"
```

Screenshot of test script being run:

```
HPING 192.168.0.19 (em1 192.168.0.19): icmp mode set, 28 headers + 0 data bytes
len=46 ip=192.168.0.19 ttl=63 id=57487 icmp_seq=0 rtt=0.9 ms
len=46 ip=192.168.0.19 ttl=63 id=57488 icmp_seq=1 rtt=0.6 ms
len=46 ip=192.168.0.19 ttl=63 id=57489 icmp_seq=2 rtt=0.6 ms
len=46 ip=192.168.0.19 ttl=63 id=57490 icmp_seq=3 rtt=0.5 ms
len=46 ip=192.168.0.19 ttl=63 id=57491 icmp_seq=4 rtt=0.6 ms

--- 192.168.0.19 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.5/0.6/0.9 ms
```

Screenshot of iptables -L -n -v -x

```
Chain icmpin (1 references)
    pkts      bytes target     prot opt in     out     source               destination
       5      140 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0           i
cmptype 0 state NEW,ESTABLISHED
       5      140 ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0           i
cmptype 8 state NEW,ESTABLISHED
```

Test Case 4

This test case was to ensure that all connections to the firewall, other than port forwarded connections, would be dropped. We sent SYN packets to a large range of ports on the firewall.

Internal Test

Excerpt from test script:

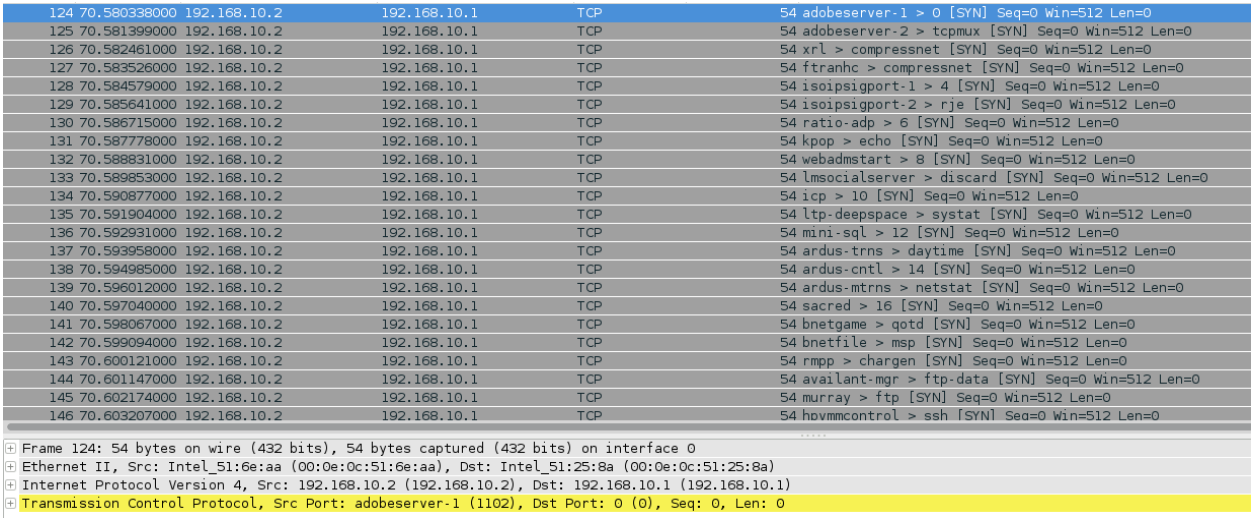
```
49 ################################# Test Case 4 ###################################
50 echo "Test case 4 commencing..."
51 CASE=4
52 # Drop all packets destined to the firewall host from outside
53
54 echo "Pinging packets from internal host to firewall:"
55 hping3 $FIREWALL_IP -p ++0 -c 2000 -i u1000 -S > $BASEFILE$CASE
56 # Refer to Test Case 1 for ssh and they are dnat'd ports
57 echo "Test case 4 results written to file"
```

Screenshot of test script being run:

```
Test case 4 commencing...
Pinging packets from internal host to firewall:

--- 192.168.10.1 hping statistic ---
2000 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 4 results written to file
```

Test output:

Screenshot of Wireshark where all packets to the firewall were dropped.



Screenshot of Dropped packets in iptables -L -n -x -v

```
Chain INPUT (policy DROP 1017 packets, 40680 bytes)
    pkts      bytes target     prot opt in     out     source               destination
    2000    80000 dhcpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
    2000    80000 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
    2000    80000 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
    1017    40680 necessitiesin  all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 DROP       udp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 0,23
       0        0 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport sports 0,23
       2       80 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 0,23
       0        0 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport sports 0:1023 state NEW
       0        0 DROP       udp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport sports 0:1023 state NEW
     976    39040 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports  !0:1023 state NEW
       0        0 DROP       udp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports  !0:1023 state NEW
       5      200 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
```

From the output, notice that no connections to the firewall were replied to and that in the iptables logs 2000 packets were dropped in the INPUT chain.

External Test

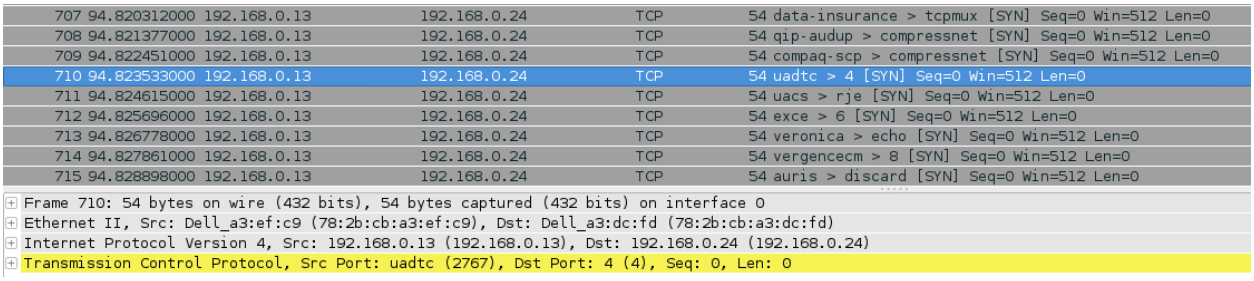
Excerpt from test script:

```
50 ################################## Test Case 4 ###################################
51 echo "Test case 4 commencing..."
52 CASE=4
53 # Drop all packets destined to the firewall host from outside
54
55 echo "Pinging packets from external host to firewall:"
56 hping3 $IP -p ++0 -c 2000 -i u1000 -S > $BASEFILE$CASE
57 # Refer to Test Case 1 for ssh and they are dnat'd ports
58 echo "Test case 4 results written to file"
```

Screenshot of test script being run:

```
Test case 4 commencing...
Pinging packets from external host to firewall:

--- 192.168.0.24 hping statistic ---
2000 packets tramitted, 3 packets received, 100% packet loss
round-trip min/avg/max = 0.5/0.6/0.7 ms
Test case 4 results written to file
```

Screenshot of Wireshark:



Test Output:

```
test4
1 HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
2 len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=22 win=29200 rtt=0.7 ms
3 len=46 ip=192.168.0.24 ttl=63 DF id=34280 sport=80 flags=RA seq=80 win=0 rtt=0.5 ms
4 len=46 ip=192.168.0.24 ttl=63 DF id=34281 sport=443 flags=RA seq=443 win=0 rtt=0.6 ms
```

From the output, notice that the only returned packets are from the externally accepted Port Forwarded ports(22,80,443) tested in Test Case 1 External Test.

Test Case 5

This test case was to ensure that all new connections destined to high ports be dropped. We sent SYN packets to ports 1024 - 6024.

Internal Test

Excerpt from test script:

```
60 ################################## Test Case 5 ###################################
61 echo "Test case 5 commencing..."
62 CASE=5
63 # Drop SYN packets to high ports
64 echo "Pinging SYN packets to high ports to host $IP:"
65 hping3 $IP -p ++1024 -c 5000 -i u1000 -S > $BASEFILE$CASE
66 echo "Test case 5 results written to file"
67
```

Screenshot of test script being run:

```
Test case 5 commencing...
Pinging SYN packets to high ports to host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5000 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 5 results written to file
```

Screenshot of Wireshark where all packets to high ports were dropped

2124	73.657376000	192.168.10.2	192.168.0.14	TCP	54 tn-tl-r1 > 1024 [SYN] Seq=0 Win=512 Len=0
2125	73.658438000	192.168.10.2	192.168.0.14	TCP	54 mil-2045-47001 > blackjack [SYN] Seq=0 Win=512 Len=0
2126	73.659490000	192.168.10.2	192.168.0.14	TCP	54 msims > cap [SYN] Seq=0 Win=512 Len=0
2127	73.660543000	192.168.10.2	192.168.0.14	TCP	54 simbaexpress > 1027 [SYN] Seq=0 Win=512 Len=0
2128	73.661613000	192.168.10.2	192.168.0.14	TCP	54 tn-tl-fd2 > 1028 [SYN] Seq=0 Win=512 Len=0
2129	73.662685000	192.168.10.2	192.168.0.14	TCP	54 intv > solid-mux [SYN] Seq=0 Win=512 Len=0
2130	73.663748000	192.168.10.2	192.168.0.14	TCP	54 ibm-abtact > iad1 [SYN] Seq=0 Win=512 Len=0
2131	73.664787000	192.168.10.2	192.168.0.14	TCP	54 pra-elmd > iad2 [SYN] Seq=0 Win=512 Len=0
2132	73.665837000	192.168.10.2	192.168.0.14	TCP	54 triquest-lm > iad3 [SYN] Seq=0 Win=512 Len=0
2133	73.666876000	192.168.10.2	192.168.0.14	TCP	54 vqp > netinfo-local [SYN] Seq=0 Win=512 Len=0
2134	73.667909000	192.168.10.2	192.168.0.14	TCP	54 gemini-lm > activesync [SYN] Seq=0 Win=512 Len=0
2135	73.668941000	192.168.10.2	192.168.0.14	TCP	54 ncpm-pm > mxxrlogin [SYN] Seq=0 Win=512 Len=0
2136	73.669970000	192.168.10.2	192.168.0.14	TCP	54 commonspace > nsstp [SYN] Seq=0 Win=512 Len=0
2137	73.671003000	192.168.10.2	192.168.0.14	TCP	54 mainsoft-lm > ams [SYN] Seq=0 Win=512 Len=0
2138	73.672035000	192.168.10.2	192.168.0.14	TCP	54 sixtrak > mtqp [SYN] Seq=0 Win=512 Len=0
2139	73.673061000	192.168.10.2	192.168.0.14	TCP	54 radio > sbl [SYN] Seq=0 Win=512 Len=0
2140	73.674088000	192.168.10.2	192.168.0.14	TCP	54 radio-sm > netarx [SYN] Seq=0 Win=512 Len=0
2141	73.675116000	192.168.10.2	192.168.0.14	TCP	54 orbplus-iiop > danf-ak2 [SYN] Seq=0 Win=512 Len=0
2142	73.676144000	192.168.10.2	192.168.0.14	TCP	54 picknfs > afrog [SYN] Seq=0 Win=512 Len=0
2143	73.677220000	192.168.10.2	192.168.0.14	TCP	54 simbaservices > boinc-client [SYN] Seq=0 Win=512 Len=0
2144	73.678275000	192.168.10.2	192.168.0.14	TCP	54 issd > dcutility [SYN] Seq=0 Win=512 Len=0
2145	73.679328000	192.168.10.2	192.168.0.14	TCP	54 aas > fpitp [SYN] Seq=0 Win=512 Len=0

```
⊞ Frame 2124: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
⊞ Ethernet II, Src: Intel_51:6e:aa (00:0e:0c:51:6e:aa), Dst: Intel_51:25:8a (00:0e:0c:51:25:8a)
⊞ Internet Protocol Version 4, Src: 192.168.10.2 (192.168.10.2), Dst: 192.168.0.14 (192.168.0.14)
⊞ Transmission Control Protocol, Src Port: tn-tl-r1 (1580), Dst Port: 1024 (1024), Seq: 0, Len: 0
```

Script Output:

test5

```
1 HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
```

Screenshot of Dropped packets in iptables -L -n -x -v

```
ltiport sports 0:1023 state NEW
    5000   200000 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
```

From the output, notice that no new connections to high ports were responded to and were dropped as shown in the iptables logs.

External Test

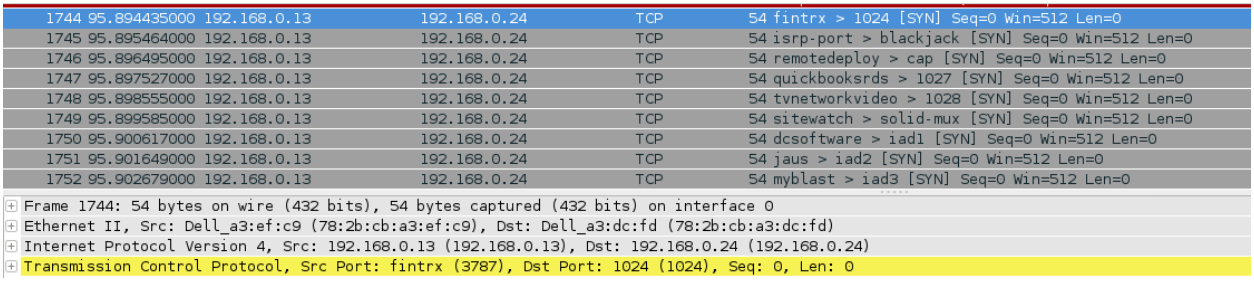
Excerpt from test script:

```
61 ################################## Test Case 5 ###################################
62 echo "Test case 5 commencing..."
63 CASE=5
64 # Drop SYN packets to high ports
65 echo "Pinging SYN packets to high ports to host $IP:"
66 hping3 $IP -p ++1024 -c 5000 -i u1000 -S > $BASEFILE$CASE
67 echo "Test case 5 results written to file"
```

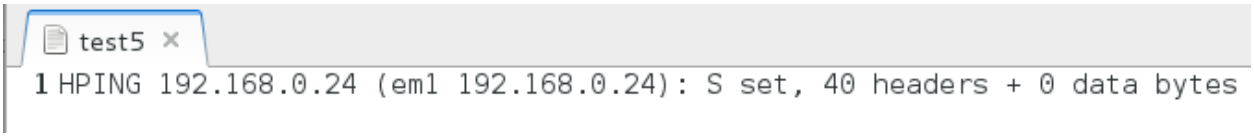
Screenshot of test script being run:

```
Test case 5 commencing...
Pinging SYN packets to high ports to host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5000 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 5 results written to file
```

Screenshot of Wireshark:



Test Output:



Screenshot of Dropped packets in iptables -L -n -x -v

From the output, notice that no new connections to high ports were successful and were dropped as shown in the iptable logs.

Test Case 6

The following test was run to have all new connections originating from low ports be dropped. We sent SYN packets to port 80 from ports 0 - 1023

Internal Test

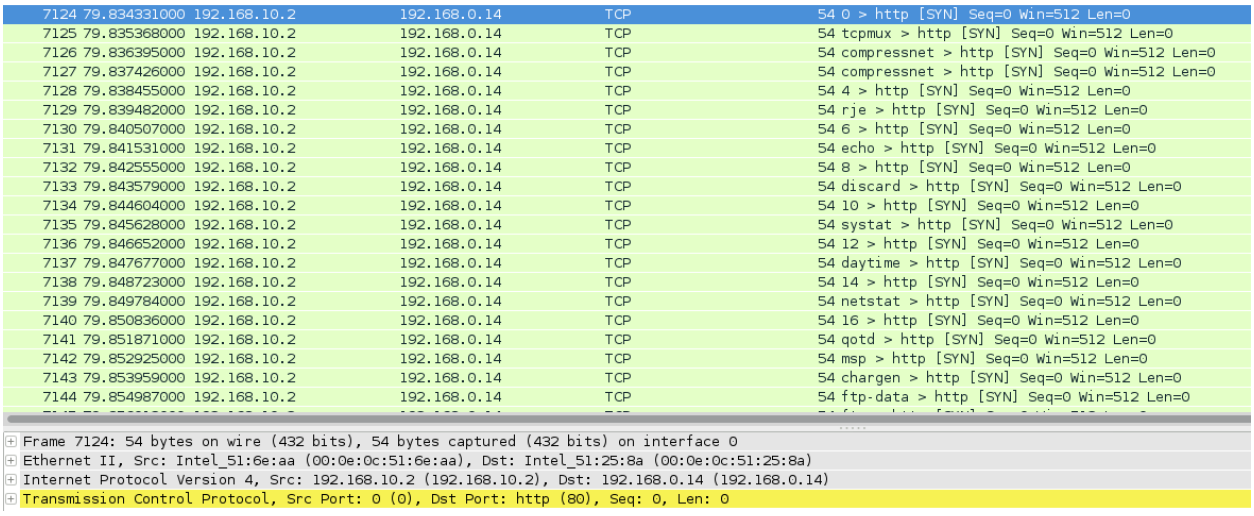
Excerpt from test script:

```
69 ################################## Test Case 6 ###################################
70 echo "Test case 6 commencing..."
71 CASE=6
72 # Drop SYN packets from low ports
73 echo "Pinging SYN packets from low ports to host $IP:"
74 hping3 $IP -s 0 -p 80 -c 1024 -i u1000 -S > $BASEFILE$CASE
75 echo "Test case 6 results written to file"
```

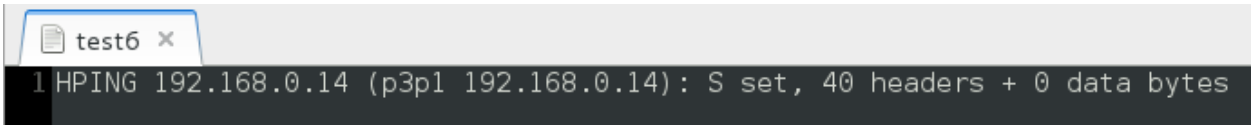
Screenshot of test script being run:

```
Test case 6 commencing...
Pinging SYN packets from low ports to host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
1024 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 6 results written to file
```

Screenshot of Wireshark where all packets to low ports are being dropped:



Test Output:



Screenshot of Dropped packets in iptables -L -n -x -v:

From the output, notice that no new connections from low ports were successful.
in the logs there were 1024 dropped packets

External Test

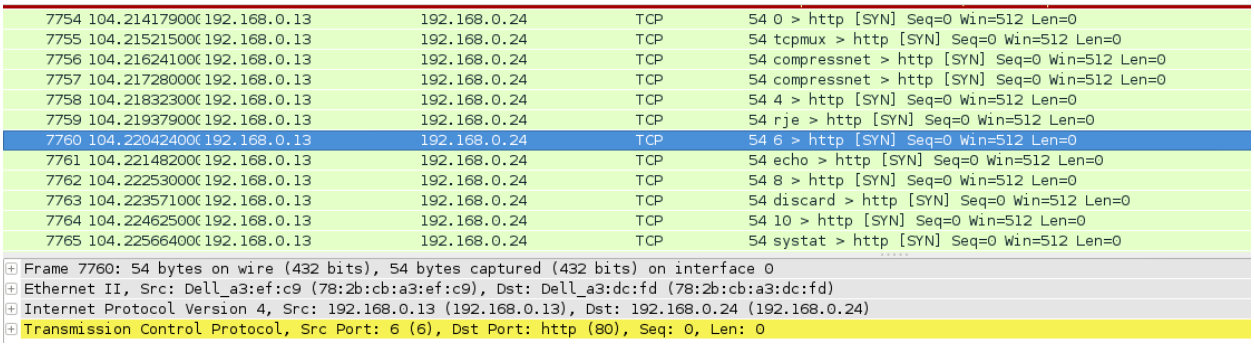
Excerpt from test script:

Screenshot of test script being run:

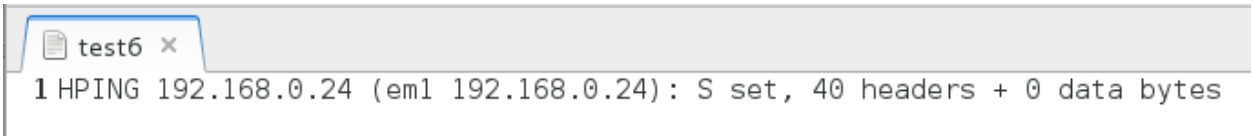
```
Test case 6 commencing...
Pinging SYN packets from low ports to host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
1024 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 6 results written to file
```

```
70 ################################## Test Case 6 ###################################
71 echo "Test case 6 commencing..."
72 CASE=6
73 # Drop SYN packets from low ports
74 echo "Pinging SYN packets from low ports to host $IP:"
75 hping3 $IP -s 0 -p 80 -c 1024 -i u1000 -S > $BASEFILE$CASE
76 echo "Test case 6 results written to file"
```

Screenshot of Wireshark:



Test Output:



Screenshot of Dropped packets in iptables -L -n -x -v

From the output, notice that no new connections from low ports were successful.
in the logs there were 1024 dropped packets

Test Case 7

The following test was run to have fragmented packets be allowed through the firewall. We sent fragmented SYN packets to port 80. From documentation, we know that iptables/netfilter will automatically aggregate fragments for us if we use stateful filtering. For this reason, no extra rule was made no rule in iptables filter table can be checked for this.

Internal Test

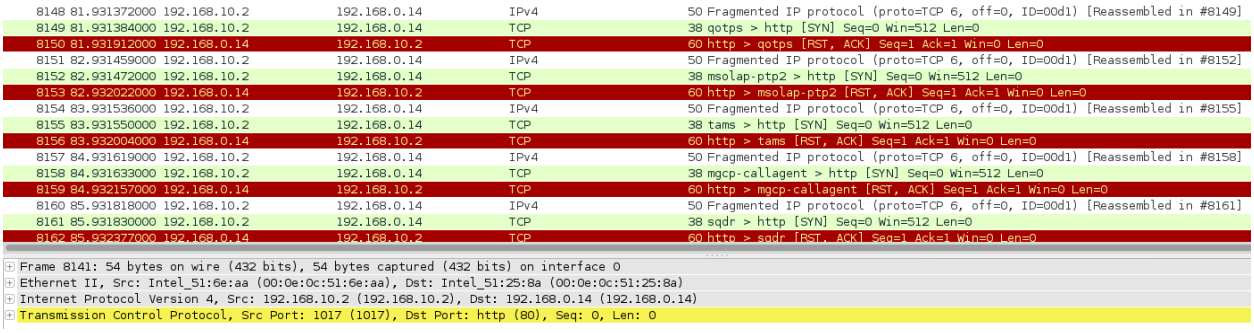
Excerpt from Test Script:

```
################################## Test Case 7 ###################################
echo "Test case 7 commencing..."
CASE=7
# fragments
echo "Sending TCP/IP fragments to port 80 of host $IP:"
hping3 $IP -S -f -p 80 -c 5 > $BASEFILE$CASE
echo "Test case 7 results written to file"
```

Screenshot of test script being run:

```
Test case 7 commencing...
Sending TCP/IP fragments to port 80 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.5/0.6/0.6 ms
Test case 7 results written to file
```

Screenshot of Wireshark:



Test Output:

test7

```
1 HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
2 len=46 ip=192.168.0.14 ttl=63 DF id=10355 sport=80 flags=RA seq=0 win=0 rtt=0.6 ms
3 len=46 ip=192.168.0.14 ttl=63 DF id=10356 sport=80 flags=RA seq=1 win=0 rtt=0.6 ms
4 len=46 ip=192.168.0.14 ttl=63 DF id=10357 sport=80 flags=RA seq=2 win=0 rtt=0.5 ms
5 len=46 ip=192.168.0.14 ttl=63 DF id=10358 sport=80 flags=RA seq=3 win=0 rtt=0.6 ms
6 len=46 ip=192.168.0.14 ttl=63 DF id=10359 sport=80 flags=RA seq=4 win=0 rtt=0.6 ms
```

From the output, notice that the fragmented SYN packet successfully passed through the firewall.

External Test

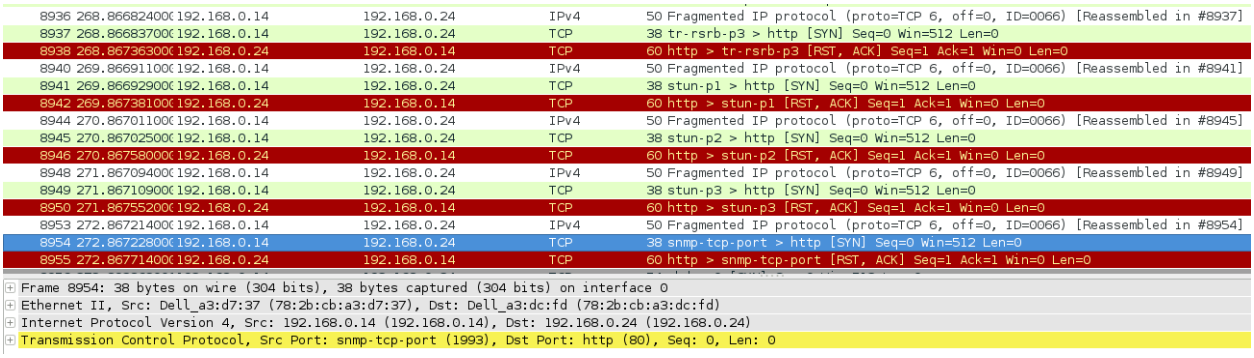
Excerpt from test script:

```
78 ################################## Test Case 7 ###################################
79 echo "Test case 7 commencing..."
80 CASE=7
81 # fragments
82 echo "Sending TCP/IP fragments to port 80 of host $IP:"
83 hping3 $IP -S -f -p 80 -c 5 > $BASEFILE$CASE
84 echo "Test case 7 results written to file"
```

Screenshot of test script being run:

```
Test case 7 commencing...
Sending TCP/IP fragments to port 80 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.6/0.6/0.7 ms
Test case 7 results written to file
```

Screenshot of Wireshark:



Test Output:

test7

```
1 HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
2 len=46 ip=192.168.0.24 ttl=63 DF id=34282 sport=80 flags=RA seq=0 win=0 rtt=0.6 ms
3 len=46 ip=192.168.0.24 ttl=63 DF id=34283 sport=80 flags=RA seq=1 win=0 rtt=0.6 ms
4 len=46 ip=192.168.0.24 ttl=63 DF id=34284 sport=80 flags=RA seq=2 win=0 rtt=0.6 ms
5 len=46 ip=192.168.0.24 ttl=63 DF id=34285 sport=80 flags=RA seq=3 win=0 rtt=0.7 ms
6 len=46 ip=192.168.0.24 ttl=63 DF id=34286 sport=80 flags=RA seq=4 win=0 rtt=0.6 ms
```

From the output, notice that the fragmented SYN packet successfully passed through the firewall

Test Case 8

The following test was run to have packets on existing connections be allowed through the firewall.

Internal Test

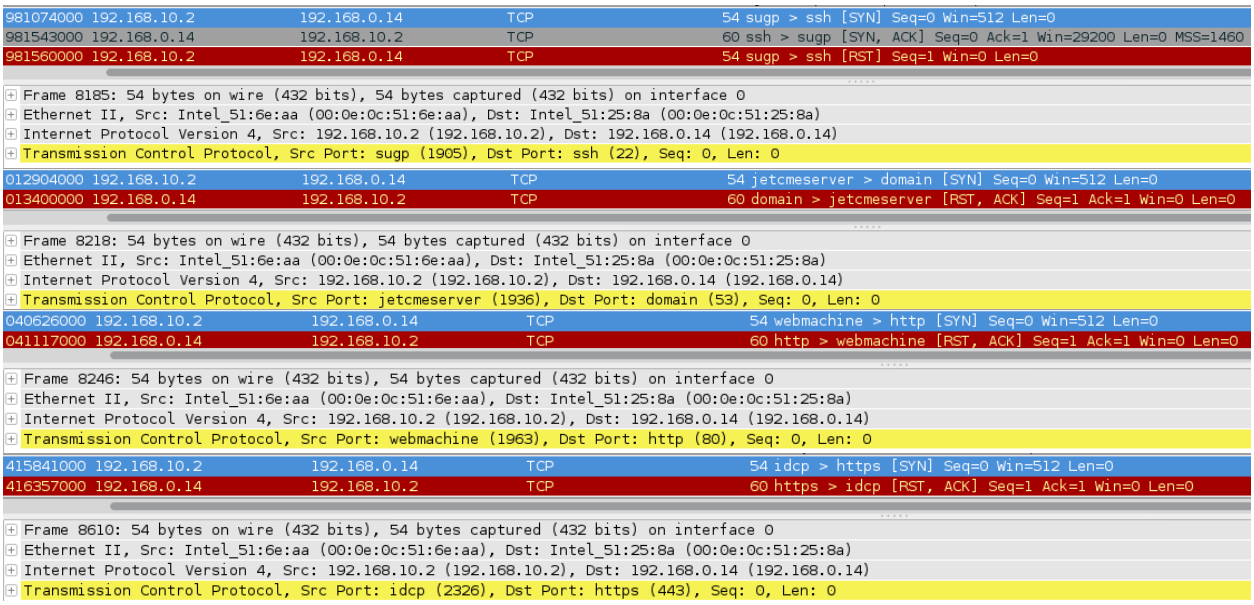
Excerpt from Test Script:

```
85 ################################## Test Case 8 ###################################
86 echo "Test case 8 commencing..."
87 CASE=8
88 #existing connections
89 echo "Sending packets to existing connections to ports 0 - 1023:"
90 hping3 $IP -p ++0 -c 1024 -i u1000 -S > $BASEFILE$CASE
91 echo "Test case 8 results written to file"
```

Screenshot of test script being run:

```
Test case 8 commencing...
Sending packets to existing connections to ports 0 - 1023:

--- 192.168.0.14 hping statistic ---
1024 packets transmitted, 4 packets received, 100% packet loss
round-trip min/avg/max = 0.5/0.5/0.5 ms
Test case 8 results written to file
```

Screenshot of Wireshark:



Test Output:

test8

```
1 HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
2 len=46 ip=192.168.0.14 ttl=63 DF id=0 sport=22 flags=SA seq=22 win=29200 rtt=0.5 ms
3 len=46 ip=192.168.0.14 ttl=63 DF id=10360 sport=53 flags=RA seq=53 win=0 rtt=0.5 ms
4 len=46 ip=192.168.0.14 ttl=63 DF id=10362 sport=80 flags=RA seq=80 win=0 rtt=0.5 ms
5 len=46 ip=192.168.0.14 ttl=63 DF id=10363 sport=443 flags=RA seq=443 win=0 rtt=0.5 ms
```

From the output, notice that the 3 way handshake successfully passed through the firewall.

External Test

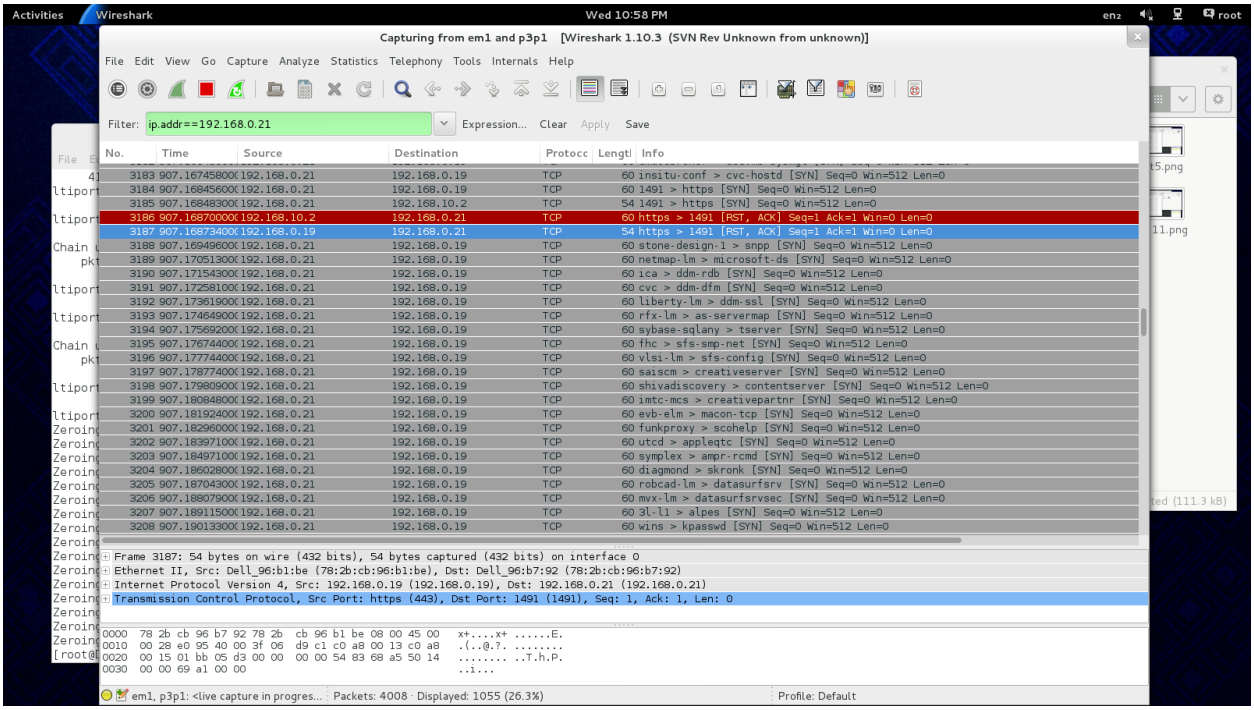
Excerpt from test script:

```
86 ################################## Test Case 8 ###################################
87 echo "Test case 8 commencing..."
88 CASE=8
89 #existing connections
90 echo "Sending packets to existing connections to ports 0 - 1023:"
91 hping3 $IP -p ++0 -c 1024 -i u1000 -S > $BASEFILE$CASE
92 echo "Test case 8 results written to file"
```

Screenshot of test script being run:

```
Test case 8 commencing...
Sending packets to existing connections to ports 0 - 1023:

--- 192.168.0.24 hping statistic ---
1024 packets tramitted, 3 packets received, 100% packet loss
round-trip min/avg/max = 0.5/0.6/0.6 ms
Test case 8 results written to file
```

Screenshot of Wireshark:



Test Output:

```
test8
1 HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
2 len=46 ip=192.168.0.24 ttl=63 DF id=0 sport=22 flags=SA seq=22 win=29200 rtt=0.6 ms
3 len=46 ip=192.168.0.24 ttl=63 DF id=34287 sport=80 flags=RA seq=80 win=0 rtt=0.5 ms
4 len=46 ip=192.168.0.24 ttl=63 DF id=34288 sport=443 flags=RA seq=443 win=0 rtt=0.5 ms
```

From the output, notice that the 3 way handshake successfully passed through the firewall

Test Case 9

The following test was run to have inbound packets with source IP matching internal network be dropped and outbound packets with source IP not matching the internal network be dropped.

Internal Test

Excerpt from Test Script:

```
94 ################################## Test Case 9 ###################################
95 echo "Test case 9 commencing..."
96 CASE=9
97 # Outside matching external network
98 hping3 $IP -S -c 5 -p 80 -a $SPOOFED_ADDR > $BASEFILE$CASE
99 echo "Test case 9 results written to file"
100
```

Screenshot of test script being run:

```
Test case 9 commencing...

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 9 results written to file
```

Test Output:

test9 ×

```
1 HPING 192.168.0.14 (p3p1 192.168.0.14): NO FLAGS are set, 40 headers + 0 data bytes
```

Notice that there were no responses in the output.
In the iptables log, the spoofed packets would make it to the NAT table but would get dropped before hitting the Filter table. We have been unable to figure out why this happens but since it is functionality that we desire anyways, we will accept this.

External Test

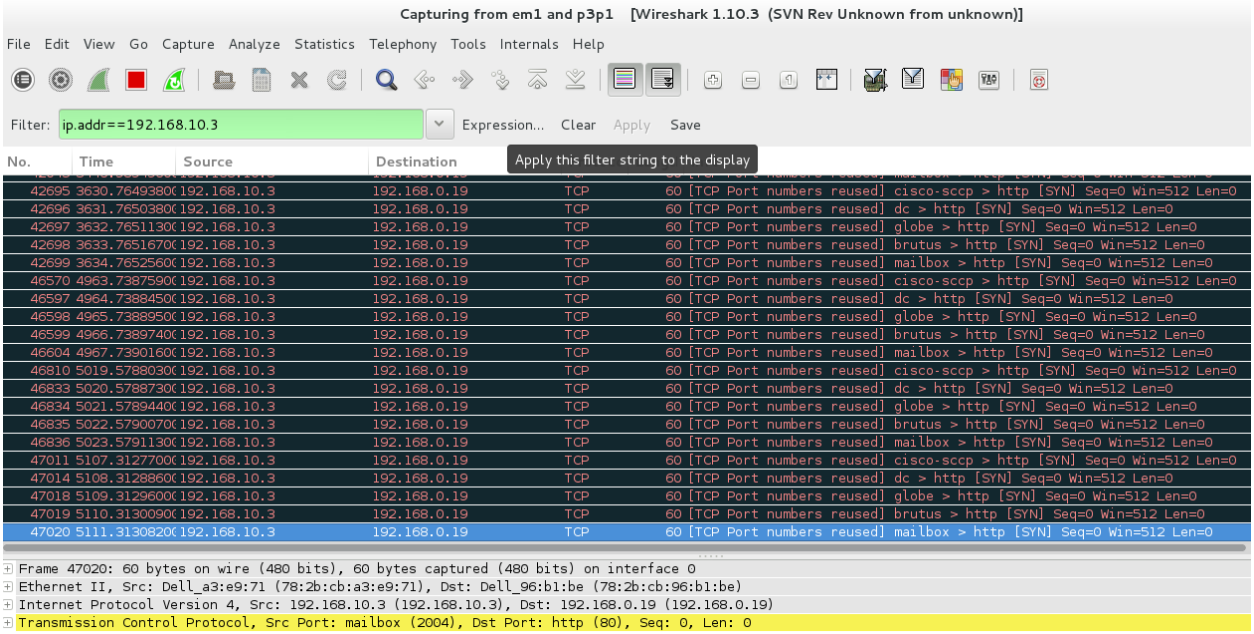
Excerpt from test script:

```
94 ################################## Test Case 9 ###################################
95 echo "Test case 9 commencing..."
96 CASE=9
97 # Outside matching internal network
98 echo "Sending packets with IP matching internal network via $SPOOFED_ADDR: "
99 hping3 $IP -S -c 5 -p 80 -a $SPOOFED_IP > $BASEFILE$CASE
.00 echo "Test case 9 results written to file"
```

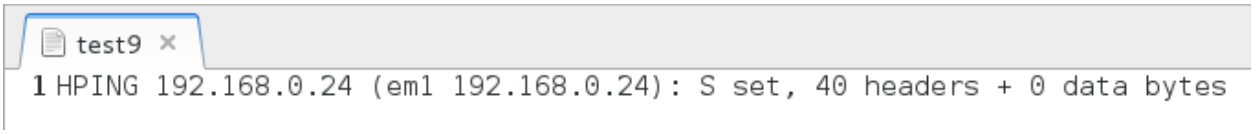
Screenshot of test script being run:

```
Test case 9 commencing...
Sending packets with IP matching internal network via :

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 9 results written to file
```

Screenshot of Wireshark:



Test Output:



Notice that there were no responses in the output.
In the iptables log, the spoofed packets would make it to the NAT table but would get dropped before hitting the Filter table. We have been unable to figure out why this happens but since it is functionality that we desire anyways, we will accept this.

iptables -L -n -v -x

```
Chain PREROUTING (policy ACCEPT 2 packets, 404 bytes)
    pkts      bytes target     prot opt in     out     source               destination
       5      200 DNAT       tcp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 80,22,443 to:192.168.10.2
```

iptables -L -n -v -x -t nat

```
Chain INPUT (policy DROP 0 packets, 0 bytes)
    pkts      bytes target     prot opt in     out     source               destination
       0        0 dhcpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 necessitiesin  all  --  *      *       0.0.0.0/0            0.0.0.0/0


Chain FORWARD (policy DROP 0 packets, 0 bytes)
    pkts      bytes target     prot opt in     out     source               destination
       0        0 dhcpforward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 necessitiesforward  all  --  *      *       0.0.0.0/0            0.0.0.0/0

       0        0 icmpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 udpin      udp  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 tcpin      tcp  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 udpout     udp  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 tcpout     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy DROP 2 packets, 404 bytes)
    pkts      bytes target     prot opt in     out     source               destination
       2      404 dhcpout    all  --  *      *       0.0.0.0/0            0.0.0.0/0
       2      404 necessitiesout  all  --  *      *       0.0.0.0/0            0.0.0.0/0
```

Test Case 10

The following test was run to have packets with both SYN and FIN flags set be dropped.

Internal Test

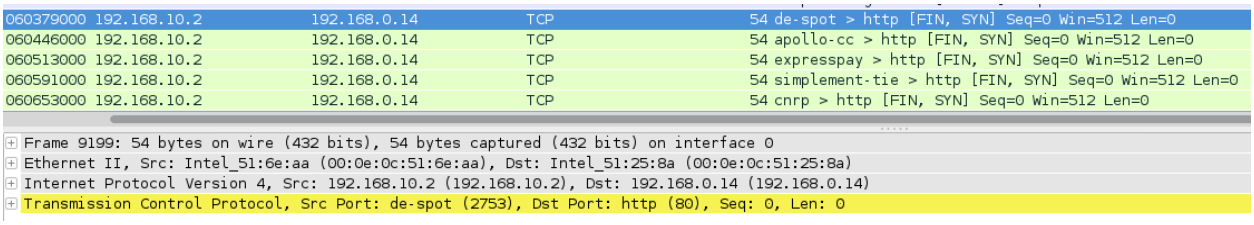
Excerpt from Test Script:

```
100 ################################# Test Case 10 ###################################
101 echo "Test case 10 commencing..."
102 CASE=10
103 #Sending Packets with SYN and FIN flags toggled
104 echo "Sending packets with SYN and FIN flags toggled to port 80 of host $IP:"
105
106 hping3 $IP -p 80 -c 5 -S -F > $BASEFILE$CASE
107 echo "Test case 10 results written to file"
```

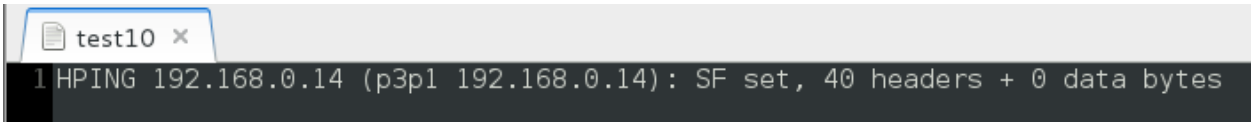
Screenshot of test script being run:

```
Test case 10 commencing...
Sending packets with SYN and FIN flags toggled to port 80 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 10 results written to file
```

Screenshot of Wireshark:



Test Output:



Screenshot of Dropped packets in iptables -L -n -v -x:

```
Chain blockout (2 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 DROP       all  --  p3p1   *      !192.168.10.0/24      0.0.0.0/0
    0     0 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           tcp flags:!0x17/0x02 state NEW
    5   200 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           tcp flags:0x03/0x03
```

Notice that the packets with SYN and FIN flags both set, were dropped in the log and that there were no responses in the output.

External Test

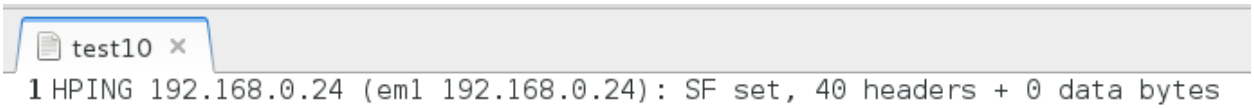
Excerpt from test script:

```
102 ################################## Test Case 10 ###################################
103 echo "Test case 10 commencing..."
104 CASE=10
105 #Sending Packets with SYN and FIN flags toggled
106 echo "Sending packets with SYN and FIN flags toggled to port 80 of host $IP:"
107
108 hping3 $IP -p 80 -c 5 -S -F > $BASEFILE$CASE
109 echo "Test case 10 results written to file"
```

Screenshot of test script being run:

```
Test case 10 commencing...
Sending packets with SYN and FIN flags toggled to port 80 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 10 results written to file
```

Test Output:



Screenshot of iptables -L -n -v -x:

```
    5      200 DROP       tcp  --  em1    *       0.0.0.0/0            0.0.0.0/0           tc
p flags:0x03/0x03
```

Notice that the packets with SYN and FIN flags both set, were dropped in the log and that there were no responses in the output.

Test Case 11

The following test was run to have all Telnet packets be dropped.

Internal Test

Excerpt from Test Script:

```
################################## Test Case 11 ###################################
echo "Test case 11 commencing..."
CASE=11
#Sending packets to port 23
echo "Sending packets to port 23 via Telnet of host $IP:"
hping3 $IP -S -c 5 -p 23 > $BASEFILE$CASE
echo "Test case 11 results written to file"
```

Screenshot of test script being run:

```
Test case 11 commencing...
Sending packets to port 23 via Telnet of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 11 results written to file
```

Screenshot of Wireshark:

```
9.858072000 192.168.10.2    192.168.0.14    TCP    54 telnet > http [SYN] Seq=0 Win=512 Len=0
5.982093000 192.168.10.2    192.168.0.14    TCP    54 tpmd > telnet [SYN] Seq=0 Win=512 Len=0
8.088327000 192.168.10.2    192.168.0.14    TCP    54 gridgen-elmd > telnet [SYN] Seq=0 Win=512 Len=0
9.088394000 192.168.10.2    192.168.0.14    TCP    54 simba-cs > telnet [SYN] Seq=0 Win=512 Len=0
00.08843200(192.168.10.2    192.168.0.14    TCP    54 aspeclmd > telnet [SYN] Seq=0 Win=512 Len=0
01.08846800(192.168.10.2    192.168.0.14    TCP    54 vistium-share > telnet [SYN] Seq=0 Win=512 Len=0
02.08852700(192.168.10.2    192.168.0.14    TCP    54 abbaccuray > telnet [SYN] Seq=0 Win=512 Len=0

⊞ Frame 7147: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
⊞ Ethernet II, Src: Intel_51:6e:aa (00:0e:0c:51:6e:aa), Dst: Intel_51:25:8a (00:0e:0c:51:25:8a)
⊞ Internet Protocol Version 4, Src: 192.168.10.2 (192.168.10.2), Dst: 192.168.0.14 (192.168.0.14)
⊞ Transmission Control Protocol, Src Port: telnet (23), Dst Port: http (80), Seq: 0, Len: 0
```

Test Output:

```
test11 ×
1 HPING 192.168.0.14 (p3p1 192.168.0.14): S set, 40 headers + 0 data bytes
```

Screenshot of Dropped packets in iptables -L -n -v -x:

```
      5      200 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 0,23
```

Notice that the telnet packets were dropped in the log and that there were no responses in the output

External Test

Excerpt from test script:

```
111 ################################# Test Case 11 ###################################
112 echo "Test case 11 commencing..."
113 CASE=11
114 #Sending packets to port 23
115 echo "Sending packets to port 23 via Telnet of host $IP:"
116 hping3 $IP -S -c 5 -p 23 > $BASEFILE$CASE
117 echo "Test case 11 results written to file"
```

Screenshot of test script being run:

```
Test case 11 commencing...
Sending packets to port 23 via Telnet of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 11 results written to file
```

Test Output:

test11 ×

```
1 HPING 192.168.0.24 (em1 192.168.0.24): S set, 40 headers + 0 data bytes
```

Screenshot of iptables -L -n -v -x:

```
      5      200 DROP       tcp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 0,23
```

Notice that the telnet packets were dropped in the log and that there were no responses in the output

Test Case 12

The following test was run to have inbound udp packets destined to ports 32768 – 32775, 137 – 139 be dropped.

Internal Test

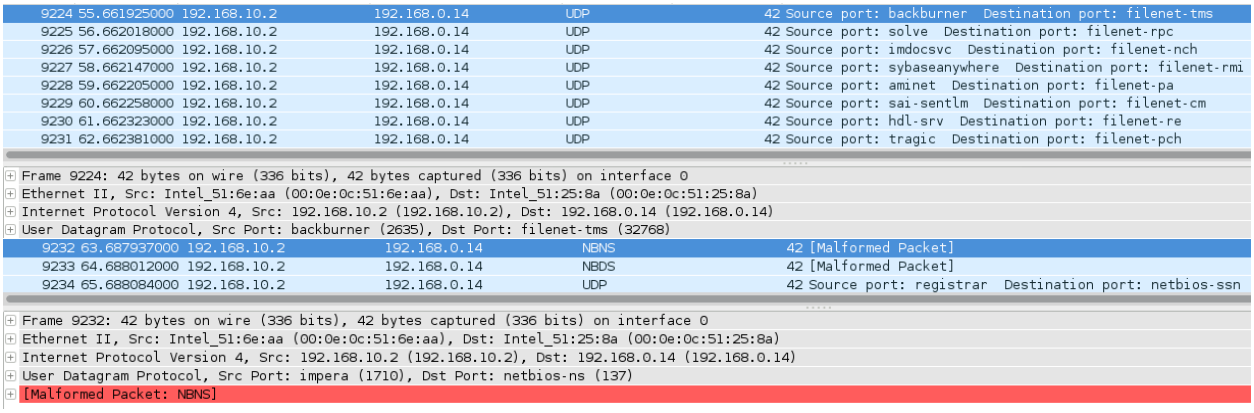
Excerpt from Test Script:

```
117 ################################# Test Case 12 ###################################
118 echo "Test case 12 commencing..."
119 CASE=12
120
121 #Sending packets to port 32768 - 32775
122 echo "Sending 8 UDP packets to port 32768 - 32775 of host $IP:"
123 hping3 $IP --udp -p 32768 -c 8 > $BASEFILE$CASE
124
125 #Sending packets to port 137 -139
126 echo "Sending 3 UDP packets to port 137 - 139 of host $IP:"
127 hping3 $IP --udp -p 137 -c 3 >> $BASEFILE$CASE
128 echo "Test case 12 results written to file"
```

Screenshot of test script being run:

```
Test case 12 commencing...
Sending 8 UDP packets to port 32768 - 32775 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
8 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 UDP packets to port 137 - 139 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
3 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 12 results written to file
```

Screenshot of Wireshark:



Screenshot of Dropped packets in iptables -L -n -v -x

```
        8      224 DROP       udp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports  !0:1023 state NEW
        0        0 DROP       tcp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 32768:32775,137:139,111,515
        3       84 DROP       udp  --  p3p1   *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 32768:32775,137:139
```

Notice that the packets were dropped in the log and that there were no responses in the output.

External Test

Excerpt from test script:

```
119 ################################## Test Case 12 ###################################
120 echo "Test case 12 commencing..."
121 CASE=12
122
123 #Sending packets to port 32768 - 32775
124 echo "Sending 8 UDP packets to port 32768 - 32775 of host $IP:"
125 hping3 $IP --udp -p ++32768 -c 8 > $BASEFILE$CASE
126
127 #Sending packets to port 137 -139
128 echo "Sending 3 UDP packets to port 137 - 139 of host $IP:"
129 hping3 $IP --udp -p ++137 -c 3 >> $BASEFILE$CASE
130 echo "Test case 12 results written to file"
```

Screenshot of test script being run:

```
Test case 12 commencing...
Sending 8 UDP packets to port 32768 - 32775 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
8 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 UDP packets to port 137 - 139 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
3 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 12 results written to file
```

Test Output:

test12

```
1 HPING 192.168.0.24 (em1 192.168.0.24): udp mode set, 28 headers + 0 data bytes
2 HPING 192.168.0.24 (em1 192.168.0.24): udp mode set, 28 headers + 0 data bytes
```

Screenshot of iptables -L -n -v -x:

```
      3      645 DROP       udp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport sports 0:1023 state NEW
      0        0 DROP       tcp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports  !0:1023 state NEW
      8      224 DROP       udp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports  !0:1023 state NEW
      0        0 DROP       tcp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 32768:32775,137:139,111,515
      3       84 DROP       udp  --  em1    *       0.0.0.0/0            0.0.0.0/0           mu
ltiport dports 32768:32775,137:139
```

Notice that the packets were dropped in the log and that there were no responses in the output.

Test Case 13

The following test was run to have inbound tcp packets destined to ports 32768 – 32775, 137 – 139, 111 and 515 be dropped.

Internal Test

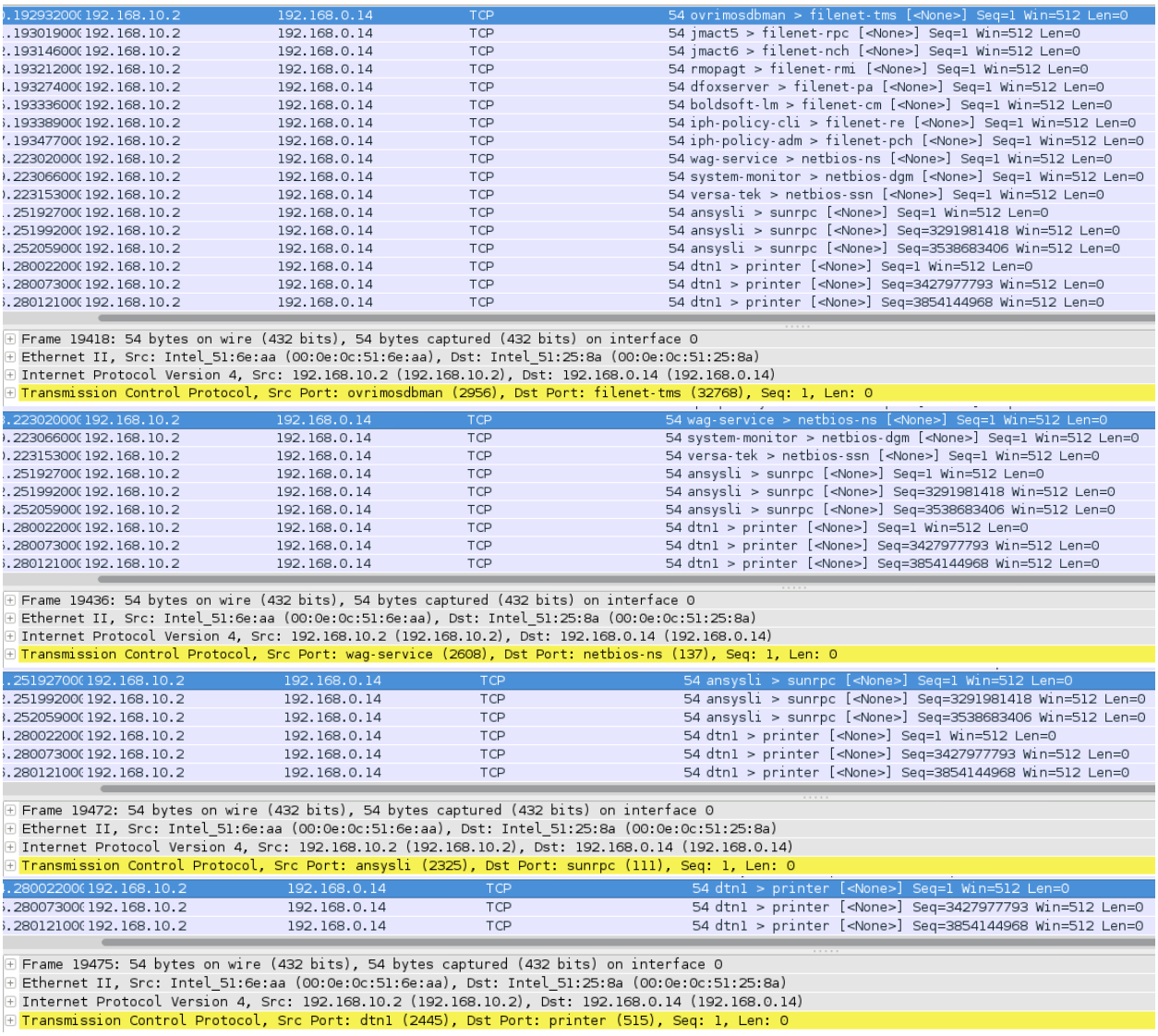
Excerpt from Test Script:

```
130 ################################# Test Case 13 ###################################
131 echo "Test case 13 commencing..."
132 CASE=13
133
134 echo "Sending 8 TCP packets to port 32768 - 32775 of host $IP:"
135 hping3 $IP -p 32768 -c 8 > $BASEFILE$CASE
136 echo "Sending 3 TCP packets to port 137 - 139 of host $IP:"
137 hping3 $IP -p 137 -c 3 >> $BASEFILE$CASE
138 echo "Sending 3 TCP packets to port 111 of host $IP:"
139 hping3 $IP -p 111 -c 3 -k >> $BASEFILE$CASE
140 echo "Sending 3 TCP packets to port 515 of host $IP:"
141 hping3 $IP -p 515 -c 3 -k >> $BASEFILE$CASE
142 echo "Test case 13 results written to file"
```

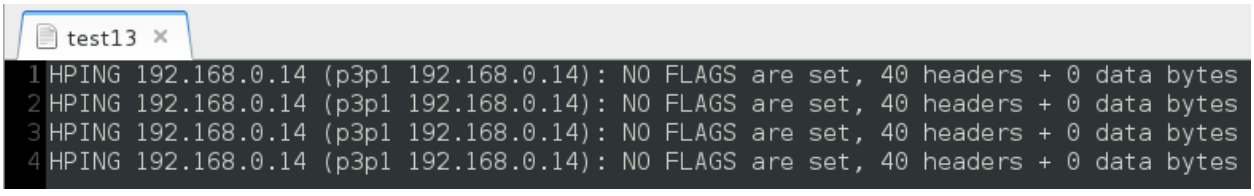
Screenshot of test script being run:

```
Test case 13 commencing...
Sending 8 TCP packets to port 32768 - 32775 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
8 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 TCP packets to port 137 - 139 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
3 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 TCP packets to port 111 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
3 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 TCP packets to port 515 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
3 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 13 results written to file
```

Screenshot of Wireshark:



Test Output:



Screenshot of Dropped packets in iptables -L -n -v -x:

Notice that the packets were dropped in the log and that there were no responses in the output.

External Test

Excerpt from test script:

```
132 ################################# Test Case 13 ###################################
133 echo "Test case 13 commencing..."
134 CASE=13
135
136 echo "Sending 8 TCP packets to port 32768 - 32775 of host $IP:"
137 hping3 $IP -p ++32768 -c 8 > $BASEFILE$CASE
138 echo "Sending 3 TCP packets to port 137 - 139 of host $IP:"
139 hping3 $IP -p ++137 -c 3 >> $BASEFILE$CASE
140 echo "Sending 3 TCP packets to port 111 of host $IP:"
141 hping3 $IP -p 111 -c 3 -k >> $BASEFILE$CASE
142 echo "Sending 3 TCP packets to port 515 of host $IP:"
143 hping3 $IP -p 515 -c 3 -k >> $BASEFILE$CASE
144 echo "Test case 13 results written to file"
```

Screenshot of test script being run:

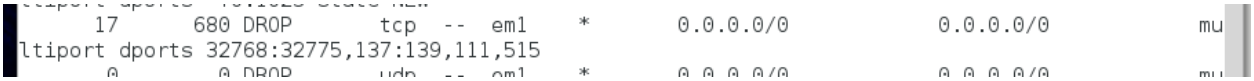
```
Test case 13 commencing...
Sending 8 TCP packets to port 32768 - 32775 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
8 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 TCP packets to port 137 - 139 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
3 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 TCP packets to port 111 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
3 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 3 TCP packets to port 515 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
3 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 13 results written to file
```

Test Output:

test13 ×

```
1 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
2 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
3 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
4 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
```

Screenshot of Dropped packets in iptables -L -n -v -x:



Notice that the packets were dropped in the log and that there were no responses in the output.

Test Case 14

The following test was run to have FTP and SSH packets be mangled with the TOS Minimize Delay flag.

Internal Test

Excerpt from Test Script:

```
################################ Test Case 14 ################################
echo "Test case 14 commencing..."

# Use other tool for FTP
CASE=14

#Mangling SSH and FTP services
echo "Mangle SSH login of host $SSH_ADDR:"
sshpass -p "uest1onQ?" ssh -o StrictHostKeyChecking=no $SSH_ADDR "ifconfig;exit" > $BASEFILE$CASE

echo "Sending 5 TCP packets to port 21 of host $IP:"
hping3 $IP -c 5 -p 21 >> $BASEFILE$CASE
echo "Sending 5 TCP packets from port 21 of host $IP:"
hping3 $IP -c 5 -s 21 -k >> $BASEFILE$CASE

echo "Test case 14 results written to file"
```

Screenshot of test script being run:

```
Test case 14 commencing...
Mangle SSH login of host root@192.168.0.14:
Sending 5 TCP packets to port 21 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 5 TCP packets from port 21 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 14 results written to file
```

Test Output. In the ssh connection, ifconfig was run to show that it is connected to the external host:

test14

```
em1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.14  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::7a2b:cbff:fea3:d737  prefixlen 64  scopeid 0x20<link>
        ether 78:2b:cb:a3:d7:37  txqueuelen 1000  (Ethernet)
        RX packets 84442  bytes 57384083 (54.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 82486  bytes 12091438 (11.5 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 20  memory 0xe1b00000-e1b20000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 504  bytes 39560 (38.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 504  bytes 39560 (38.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

HPING 192.168.0.14 (p3p1 192.168.0.14): NO FLAGS are set, 40 headers + 0 data bytes
HPING 192.168.0.14 (p3p1 192.168.0.14): NO FLAGS are set, 40 headers + 0 data bytes
```

Screenshot of mangled packets in iptables -L -n -x -v -t mangle

```
[root@DataComm standalone-fw]# iptables -L -v -t mangle
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport sports ftp,ssh  TOS setMinimize-Delay
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport sports ftp-data  TOS setMaximize-Throughput
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport dports ftp,ssh  TOS setMinimize-Delay
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport dports ftp-data  TOS setMaximize-Throughput

Chain PREROUTING (policy ACCEPT 29907 packets, 1783885 bytes)
    pkts      bytes target     prot opt in     out     source               destination
     141    27054 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 21,22 TOS set 0x10/0x3f
      18      720 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 20 TOS set 0x08/0x3f
     168    23934 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 21,22 TOS set 0x10/0x3f
      21      840 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 20 TOS set 0x08/0x3f
```

Notice that the packets were mangled in the iptables mangle log.

External Test

Excerpt from test script:

```
################################## Test Case 14 ###################################
echo "Test case 14 commencing..."

# Use other tool for FTP
CASE=14

#Mangling SSH and FTP services
echo "Mangle SSH login of host $SSH_ADDR:"
sshpass -p "uestionQ?" ssh -o StrictHostKeyChecking=no $SSH_ADDR "ifconfig;exit" > $BASEFILE$CASE

echo "Sending 5 TCP packets to port 21 of host $IP:"
hping3 $IP -c 5 -p 21 >> $BASEFILE$CASE
echo "Sending 5 TCP packets from port 21 of host $IP:"
hping3 $IP -c 5 -s 21 -k >> $BASEFILE$CASE

echo "Test case 14 results written to file"
```

Screenshot of test script being run:

```
Test case 14 commencing...
Mangle SSH login of host root@192.168.0.24:
Sending 5 TCP packets to port 21 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 5 TCP packets from port 21 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 14 results written to file
```

Test Output:

test14

```
 1 lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
 2         inet 127.0.0.1  netmask 255.0.0.0
 3         inet6 ::1  prefixlen 128  scopeid 0x10<host>
 4         loop  txqueuelen 0  (Local Loopback)
 5         RX packets 8135  bytes 709387 (692.7 KiB)
 6         RX errors 0  dropped 0  overruns 0  frame 0
 7         TX packets 8135  bytes 709387 (692.7 KiB)
 8         TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
 9
10 p3p1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
11         inet 192.168.10.2  netmask 255.255.255.0  broadcast 192.168.10.255
12         inet6 fe80::20e:cff:fe51:6eaa  prefixlen 64  scopeid 0x20<link>
13         ether 00:0e:0c:51:6e:aa  txqueuelen 1000  (Ethernet)
14         RX packets 14998  bytes 11929332 (11.3 MiB)
15         RX errors 0  dropped 0  overruns 0  frame 0
16         TX packets 22470  bytes 3145827 (3.0 MiB)
17         TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
18
19 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
20 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
```

Screenshot of mangled packets in iptables -L -n -x -v -t mangle

```
Chain PREROUTING (policy ACCEPT 29907 packets, 1783885 bytes)
    pkts      bytes target     prot opt in     out     source               destination
     141    27054 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 21,22 TOS set 0x10/0x3f
      18      720 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 20 TOS set 0x08/0x3f
     168    23934 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 21,22 TOS set 0x10/0x3f
      21      840 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 20 TOS set 0x08/0x3f
```

Notice that the packets were mangled in the log

Test Case 15

The following test was run to have FTP data packets be mangled with the TOS Maximize Throughput flag.

Internal Test

Excerpt from Test Script:

```
161 ################################## Test Case 15 ###################################
162 echo "Test case 15 commencing..."
163 # Use other tool for FTP
164 CASE=15
165
166 echo "Sending 5 TCP packets to port 20 of host $IP:"
167 hping3 $IP -c 5 -p 20 > $BASEFILE$CASE
168 echo "Sending 5 TCP packets from port 20 of host $IP:"
169 hping3 $IP -c 5 -s 20 -k >> $BASEFILE$CASE
170
171 echo "Test case 15 results written to file"
```

Screenshot of test script being run:

```
Test case 15 commencing...
Sending 5 TCP packets to port 20 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 5 TCP packets from port 20 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 15 results written to file
```

Test Output:

test15

```
1 HPING 192.168.0.14 (p3p1 192.168.0.14): NO FLAGS are set, 40 headers + 0 data bytes
2 HPING 192.168.0.14 (p3p1 192.168.0.14): NO FLAGS are set, 40 headers + 0 data bytes
```

Screenshot of mangled packets in iptables -L -n -x -v -t mangle:

```
[root@DataComm standalone-fw]# iptables -L -v -t mangle
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport sports ftp,ssh  TOS setMinimize-Delay
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport sports ftp-data  TOS setMaximize-Throughput
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport dports ftp,ssh  TOS setMinimize-Delay
    0     0 TOS        tcp  --  any    any     anywhere             anywhere            multiport dports ftp-data  TOS setMaximize-Throughput

Chain PREROUTING (policy ACCEPT 29907 packets, 1783885 bytes)
    pkts      bytes target     prot opt in     out     source               destination
     141    27054 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 21,22 TOS set 0x10/0x3f
      18      720 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 20 TOS set 0x08/0x3f
     168    23934 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 21,22 TOS set 0x10/0x3f
      21      840 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 20 TOS set 0x08/0x3f
```

Notice that the packets were mangled in the iptables mangle log

External Test

Excerpt from test script:

```
163 ################################# Test Case 15 ###################################
164 echo "Test case 15 commencing..."
165 # Use other tool for FTP
166 CASE=15
167
168 echo "Sending 5 TCP packets to port 20 of host $IP:"
169 hping3 $IP -c 5 -p 20 > $BASEFILE$CASE
170 echo "Sending 5 TCP packets from port 20 of host $IP:"
171 hping3 $IP -c 5 -s 20 -k >> $BASEFILE$CASE
172
173 echo "Test case 15 results written to file"
```

Screenshot of test script being run:

```
Test case 15 commencing...
Sending 5 TCP packets to port 20 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Sending 5 TCP packets from port 20 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 15 results written to file
```

Test Output:

test15 ×

```
1 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
2 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
```

Screenshot of mangled packets in iptables -L -n -x -v -t mangle:

```
Chain PREROUTING (policy ACCEPT 29907 packets, 1783885 bytes)
    pkts      bytes target     prot opt in     out     source               destination
     141    27054 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 21,22 TOS set 0x10/0x3f
      18      720 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport sports 20 TOS set 0x08/0x3f
     168    23934 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 21,22 TOS set 0x10/0x3f
      21      840 TOS        tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           multiport dports 20 TOS set 0x08/0x3f
```

Notice that the packets were mangled in the log

Test Case 16

The following test was run to have all other packets outside of defined rules be dropped.

Internal Test

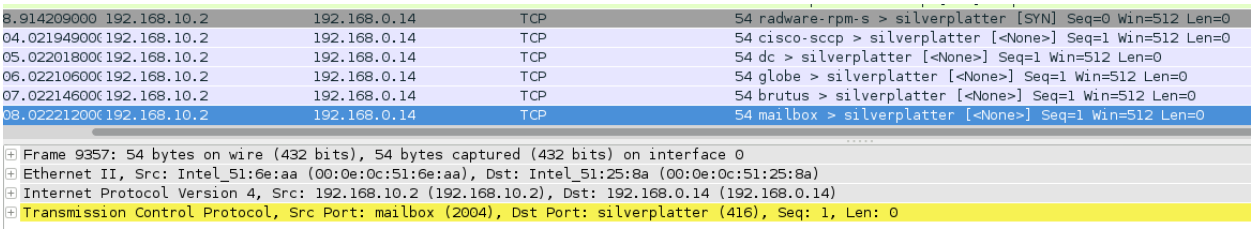
Excerpt from Test Script:

```
173 ################################## Test Case 16 ###################################
174 echo "Test case 16 commencing..."
175 CASE=16
176
177 echo "Sending 5 packets to port 416 of host $IP:"
178 hping3 $IP -s 2000 -p 416 -c 5 > $BASEFILE$CASE
179 echo "Test case 16 results written to file"
```

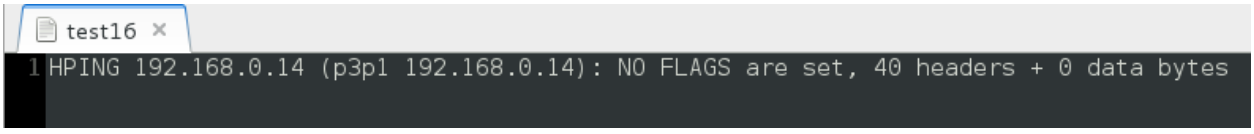
Screenshot of test script being run:

```
Test case 16 commencing...
Sending 5 packets to port 416 of host 192.168.0.14:

--- 192.168.0.14 hping statistic ---
5 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 16 results written to file
```

Screenshot of Wireshark:



Test Output:



Screenshot of Dropped packets in iptables -L -n -v -x:

```
Chain INPUT (policy DROP 29 packets, 3020 bytes)
    pkts      bytes target     prot opt in     out     source               destination
     100    16036 dhcpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
      88    12100 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
      50     6357 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
      50     6357 necessitiesin  all  --  *      *       0.0.0.0/0            0.0.0.0/0


Chain FORWARD (policy DROP 0 packets, 0 bytes)
    pkts      bytes target     prot opt in     out     source               destination
   89316 47448043 dhcpforward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
   89316 47448043 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
   89316 47448043 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
   89312 47447739 necessitiesforward  all  --  *      *       0.0.0.0/0            0.0.0.0/0

   87404 47248792 icmpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 udpin      udp  --  *      *       0.0.0.0/0            0.0.0.0/0
   87404 47248792 tcpin      tcp  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 udpout     udp  --  *      *       0.0.0.0/0            0.0.0.0/0
   42805  9038634 tcpout     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy DROP 1122 packets, 99746 bytes)
    pkts      bytes target     prot opt in     out     source               destination
    1143   101118 dhcpout    all  --  *      *       0.0.0.0/0            0.0.0.0/0
    1143   101118 necessitiesout  all  --  *      *       0.0.0.0/0            0.0.0.0/0
```

Notice the packets that did not hit any rules were dropped in the iptables logs.

External Test

Excerpt from test script:

```
175 ################################## Test Case 16 ###################################
176 echo "Test case 16 commencing..."
177 CASE=16
178
179 echo "Sending 5 packets to port 416 of host $IP:"
180 hping3 $IP -s 2000 -p 416 -c 5 > $BASEFILE$CASE
181 echo "Test case 16 results written to file"
```

Screenshot of test script being run:

```
Test case 16 commencing...
Sending 5 packets to port 416 of host 192.168.0.24:

--- 192.168.0.24 hping statistic ---
5 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
Test case 16 results written to file
```

Test Output:

test16

```
1 HPING 192.168.0.24 (em1 192.168.0.24): NO FLAGS are set, 40 headers + 0 data bytes
```

Screenshot of iptables -l -n -v -x

```
Chain INPUT (policy DROP 29 packets, 3020 bytes)
    pkts      bytes target     prot opt in     out     source               destination
     100    16036 dhcpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
      88    12100 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
      50     6357 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
      50     6357 necessitiesin  all  --  *      *       0.0.0.0/0            0.0.0.0/0


Chain FORWARD (policy DROP 0 packets, 0 bytes)
    pkts      bytes target     prot opt in     out     source               destination
   89316 47448043 dhcpforward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
   89316 47448043 blockin    all  --  *      *       0.0.0.0/0            0.0.0.0/0
   89316 47448043 blockout   all  --  *      *       0.0.0.0/0            0.0.0.0/0
   89312 47447739 necessitiesforward  all  --  *      *       0.0.0.0/0            0.0.0.0/0

   87404 47248792 icmpin     all  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 udpin      udp  --  *      *       0.0.0.0/0            0.0.0.0/0
   87404 47248792 tcpin      tcp  --  *      *       0.0.0.0/0            0.0.0.0/0
       0        0 udpout     udp  --  *      *       0.0.0.0/0            0.0.0.0/0
   42805  9038634 tcpout     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy DROP 1122 packets, 99746 bytes)
    pkts      bytes target     prot opt in     out     source               destination
    1143   101118 dhcpout    all  --  *      *       0.0.0.0/0            0.0.0.0/0
    1143   101118 necessitiesout  all  --  *      *       0.0.0.0/0            0.0.0.0/0
```

Notice the packets that did not hit any rules were dropped in the iptables logs.

Conclusion

In conclusion, the standalone firewall had a lot of rules and constraints to implement in order to meet the requirements for filtering and forwarding the proper packets from the internal hosts to the external hosts and vice versa through user-defined chains. To test these rules and constraints for the firewall to be considered robust, we use tools such as **Hping** and **Wireshark**.