COMP 8006 Assignment 2

Network Security Administration 2

SSH Monitor Application

By: Derek Wong (A01042588)

Instructor: Aman Abdulla

Due: 1 PM on February 25, 2021

Table of Contents

Objectives

Design, implement, and test a simple monitor application that will detect password-guessing attempts against the SSH server and block that IP using Netfilter:

- Application will monitor the **/var/log/secure** file (keep in mind different distributions will have different formats) and detect password-guessing attempts and then use iptables to block that IP.
- Application will get user specified parameters and continuously monitor the log file specified.
- As soon as monitor detects that the number of attempts from a particular IP has gone over a user-specified threshold, it will generate a rule to block that IP.
- If the user has specified a time limit for a block, your application will flush the rule from Firewall rule set upon expiration of the block time limit.
- Design test procedure that tests application under variety of conditions.

Approach

The application will be implemented using a python script involving subprocesses that invoke bash shell, commands *iptables* and *tail* commands to create the firewall filter and to continuously monitor firewall.

From the command line, the number of attempts before blocking the IP and the time limit in seconds for blocking the IP will be given as arguments. If the number of attempts is not specified or invalid, a default value will be used instead. In the case of time limit, an indefinite time value will be enforced, and the firewall filter will not expire.

To test the functionality of the program, we will vary the specified command line values and attack the server with multiple clients simultaneously. During these tests, wireshark will be used to analyze the inbound and outbound traffic between the server running the firewall rules and the client making the connection attempts. Wireshark captures and iptables log on server host will confirm the functionality of the firewall and whether it meets the requirements.

Test Environment Network Architecture

The testbed will have one machine operating as a server which also serves as the firewall. Two clients will be used as attacking machines targeting the server with unsuccessful password attempts.

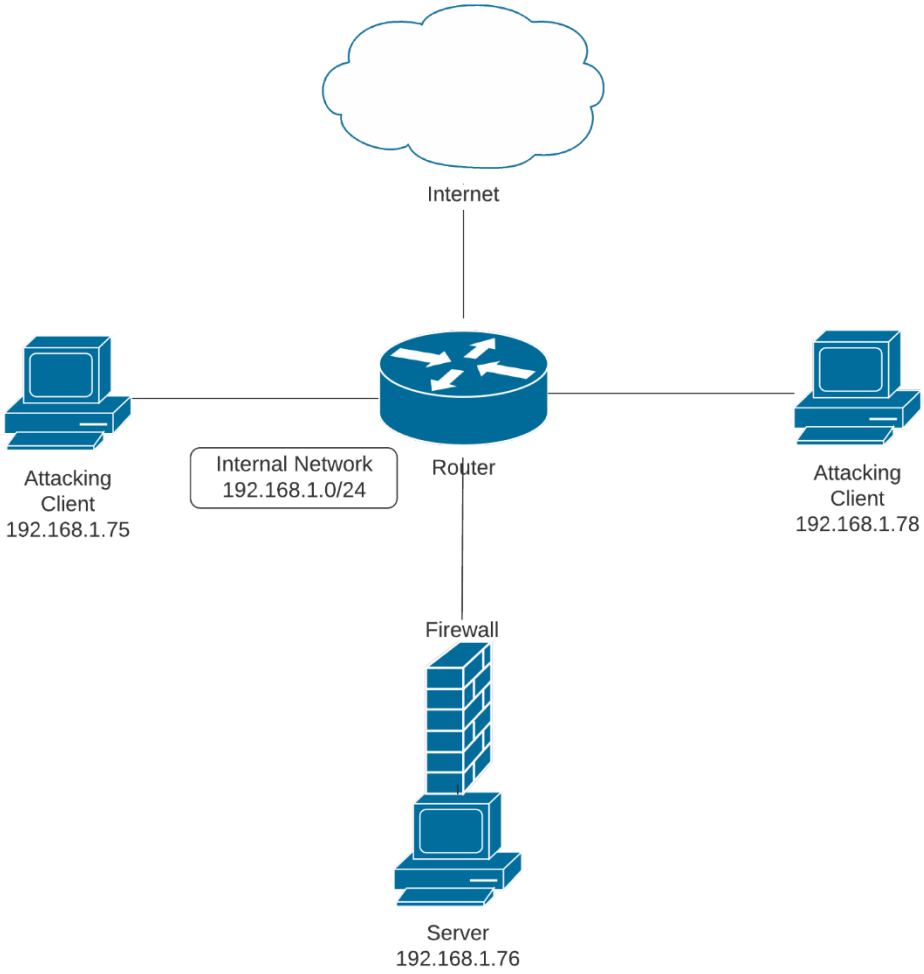


Figure 1: Network Architecture of test environment

Script Design

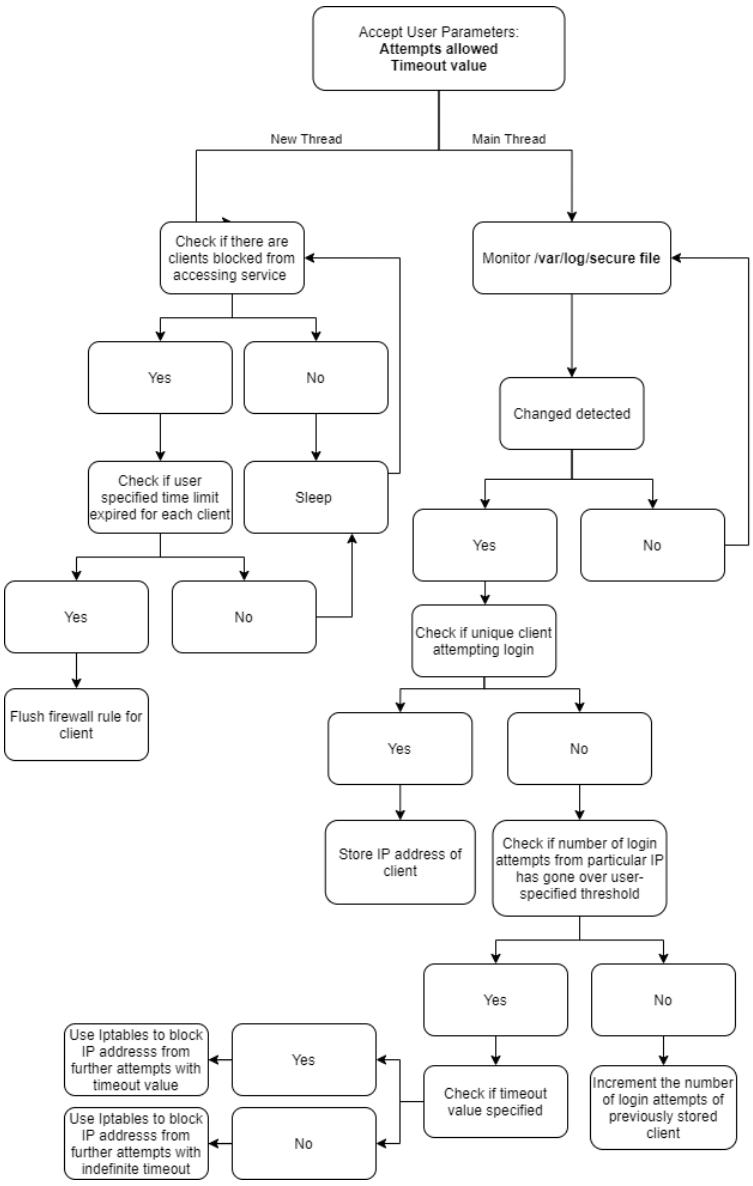


Figure 2: Application logic flow chart

Pseudocode

```
global var failed client attempts{}
FAILED_MATCH_STRING = “Failed password”

usage():
        print help for program usage

check_blocked_clients(num_attempts):
        while true:
                for ip address in failed client attempts:
                        if client is not timeout indefinitely and client exceeded maximum attempts:
                                if timer expired:
                                        remove iptables filter
                                        remove ip from failed client attempts
                sleep(polling rate)

check_secure_logs(num_attempts, timeout_sec):
        while true:
                poll and monitor the /var/log/secure file
                if line logged:
                        max_login_handler(num_attetmpts, timeout_sec)

max_login_handler(line, num_attempts, timeout_sec):
        match regex for FAILED_MATCH_STRING
        if match:
                ip_addr = obtain substring for IP address
                if ip_addr in failed client attempts:
                        if num login attempts == num_attempts:
                                if timeout_sec is not indefinite:
                                        update failed client attempts to include blocking time
                        drop all incoming traffic to ip_addr
                increment client failed attempts by ip_addr by 1
                else:
                        create entry in failed client attempts and initialize failed attempts to 1

main:
        validate running with superuser privileges
        initialize num_attempts, timeout_sec to default values
        get command line arguments for num_attempts, and timeout_sec

        create daemon thread(target=check_blocked_clients(num_attempts))
        check_secure_logs(num_attempts, timeout_sec)

        if KeyboardInterrupt:
                flush iptables rules
```

User Manual

Background

The monitoring script, **ssh_mon.py**, was executed on Linux Fedora 32 running on a Raspberry Pi 3 B+ (Quad Core).

Program Usage

$. ./ssh_mon.py [-h] -a <num_attempts> -t <timeout_sec>

Example

$. ./ssh_mon.py -a 2 -t 15

This script will allow clients a maximum of 2 attempts before creating a filter rule that will prevent the client from attempting access to the server for a timeout duration of 15 seconds.

If either of the arguments are given invalid parameters, default values will be used instead (see Important Macros).

Important Macros

```
# Symbolic Constants
DEFAULT_ATTEMPTS_ALLOWED = 3
INDEFINITE_TIMEOUT_SECONDS = "INDEFINITE"
SECURE_LOG_FILE = "/var/log/secure"
FAILED_MATCH_STRING = "Failed password"
BLOCKED_CLIENT_ATTEMPTS_KEY = "num_attempts"
BLOCKED_CLIENT_TIMEOUT_KEY = "timeout_end"
BLOCKED_CLIENT_POLLING_RATE = 0.3
DATTIME_FORMAT = "%H:%M:%S"
```

DEFAULT_ATTEMPTS_ALLOWED = 3

This setting allows you to specify the maximum unauthorized attempts to the server.

BLOCKED_CLIENT_POLLING_RATE = 0.3

This setting allows you to specify the polling rate of the monitoring function to see when a blocked client's timeout has expired.

Testing Design

Network Settings	Configuration
SSH Server	192.168.1.76
Attacking Client 1	192.168.1.75
Attacking Client 2	192.168.1.78

Test Cases

Case #	Test Description	Tool Used	Expected Result	Pass/Failed
1	Verify SSH server enforces default number of attempts when monitor is run with invalid number of attempt argument. Attacking Client 1 will send unauthorized attempts to connect with SSH server that exceed maximum attempts allowed.	wireshark, iptables	SSH server shows unauthorized attempts to login with failed password in /var/log/secure file. Once unauthorized attempts exceed the maximum attempts allowed, Iptables is populated with filter to block all traffic coming from attacking client 1 to SSH server for specified timeout in seconds. After specified timeout is reached, the filter rule is flushed. Attacking client may reconnect. Packet capture on SSH server should show received packets from attacking client 1 but packets indicating authentication failure. Packet capture on attacking client 1 should shows received packets from SSH server but no established SSH conversation.	Pass. Details are attached below in confirmatory data.
2	Verify SSH server enforces default timeout in seconds when monitor is run with invalid timeout in seconds argument. Attacking Client 1 will send unauthorized attempts to connect with SSH server that exceed maximum attempts allowed.	wireshark, iptables	SSH server shows unauthorized attempts to login with failed password in /var/log/secure file. Once unauthorized attempts exceed the maximum attempts allowed, Iptables is populated with filter to block all traffic coming from attacking client 1 to SSH server for an indefinite time. Packet capture on SSH server should show received packets from attacking client 1 but packets indicating authentication failure. Packet capture on attacking client 1 should shows received packets from SSH server but no established SSH conversation.	Pass. Details are attached below in confirmatory data.

3	Verify SSH server enforces user specified number of attempts allowed and timeout in seconds when monitor is run with valid arguments. Attacking Client 1 will send unauthorized attempts to connect with SSH server that exceed maximum attempts allowed.	wireshark, iptables	SSH server shows unauthorized attempts to login with failed password in /var/log/secure file. Once unauthorized attempts exceed the maximum attempts allowed, Iptables is populated with filter to block all traffic coming from attacking client 1 to SSH server for specified timeout in seconds. After specified timeout is reached, the filter rule is flushed. Attacking client may reconnect. Packet capture on SSH server should show received packets from attacking client 1 but packets indicating authentication failure. Packet capture on attacking client 1 should shows received packets from SSH server but no established SSH conversation.	Pass. Details are attached below in confirmatory data.
4	Verify SSH server enforces command line specified parameters to multiple simultaneous attacking clients. Attacking Client 1 and 2 will send unauthorized attempts to connect with SSH server that exceed maximum attempts allowed.	wireshark, iptables	SSH server shows unauthorized attempts to login with failed password in /var/log/secure file. Once unauthorized attempts exceed the maximum attempts allowed, Iptables is populated with filter to block all traffic coming from attacking client 1 and client 2 to SSH server for specified timeout in seconds. After specified timeout is reached, the filter rule is flushed. Attacking clients may reconnect. Packet capture on SSH server should show received packets from attacking client 1 but packets indicating authentication failure. Packet capture on attacking client 1 should shows received packets from SSH server but no established SSH conversation.	Pass. Details are attached below in confirmatory data.

Confirmatory Data

Case 1

- For this test, the following arguments are given to the monitoring program along with the associated unauthorized attempts to connect to the server and measures taken. An invalid argument is given to the maximum attempts parameter, so the monitoring program defaults to a value of 3.

```
[root@localhost Assignment-2]# ./ssh-mon.py -a invalid -t 15
Number of attempts value invalid: Default value of 3 used
Timeout of 15 seconds will be given after number of allowed attempts exceeded
Feb 16 08:34:35 localhost sshd[11767]: Failed password for radiant from 192.168.1.75 port 45846 ssh2

Feb 16 08:34:42 localhost sshd[11767]: Failed password for radiant from 192.168.1.75 port 45846 ssh2

Feb 16 08:34:48 localhost sshd[11767]: Failed password for radiant from 192.168.1.75 port 45846 ssh2

Feb 16 08:35:05 localhost sshd[11769]: Failed password for radiant from 192.168.1.75 port 45848 ssh2

Blocking IP: 192.168.1.75 for 15 seconds
Blocking IP: 192.168.1.75       Time: 08:35:05
Unblocking IP: 192.168.1.75     Time: 08:35:20
Thread Unblocked at 08:35:20
```

- Before the test, iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

- During the test when attacker 1 exceeds maximum allowed attempts of 4 , iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  192.168.1.75         0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

Iptables filters all inbound traffic to the server. This confirms the working functionality to verify the test case.

- After the timeout value of 15 seconds, iptables shows:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

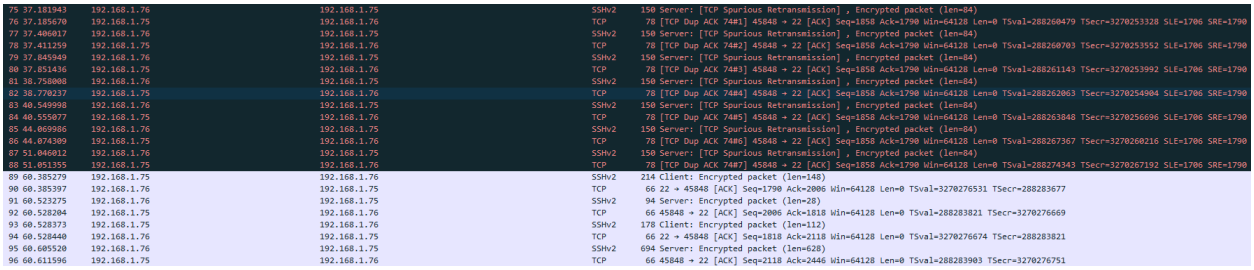
All rules are flushed, this confirms that our monitoring script was able to successfully remove the filter after the timeout completes.

- The log file /var/log/secure shows:

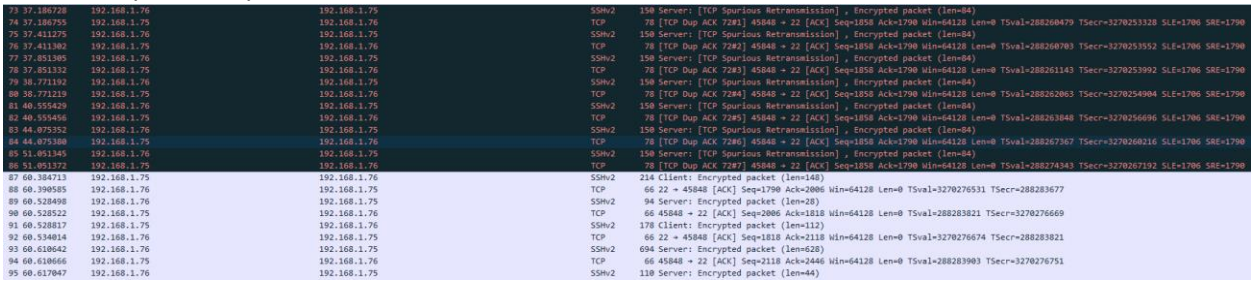
```
Feb 16 08:34:33 localhost sshd[11767]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 08:34:35 localhost sshd[11767]: Failed password for radiant from 192.168.1.75 port 45846 ssh2
Feb 16 08:34:42 localhost sshd[11767]: Failed password for radiant from 192.168.1.75 port 45846 ssh2
Feb 16 08:34:48 localhost sshd[11767]: Failed password for radiant from 192.168.1.75 port 45846 ssh2
Feb 16 08:34:50 localhost sshd[11767]: Connection closed by authenticating user radiant 192.168.1.75 port 45846 [preauth]
Feb 16 08:34:50 localhost sshd[11767]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 08:35:02 localhost sshd[11769]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 08:35:05 localhost sshd[11769]: Failed password for radiant from 192.168.1.75 port 45848 ssh2
Feb 16 08:35:30 localhost sshd[11769]: Accepted password for radiant from 192.168.1.75 port 45848 ssh2
Feb 16 08:35:30 localhost sshd[11769]: pam_unix(sshd:session): session opened for user radiant by (uid=0)
```

After the timeout, the attacking client may reconnect to the server

- Server packet capture



- Attacker 1 packet capture



These packet captures show the retransmissions occurring from the timeout events happening due to the packets being dropped at the filter (lasting approximately 15 seconds). After the timeout is exceeded, the firewall rule is flushed allowing the attacking client to attempt connections again.

Case 2

- For this test, the following arguments are given to the monitoring program along with the associated unauthorized attempts to connect to the server and measures taken. An invalid argument is given to the timeout in seconds, so the monitoring program defaults to an indefinite timeout.

```
[root@localhost Assignment-2]# ./ssh-man.py -a 2 -t invalid
Number of allowed attempts for each ssh client before timeout: 2
Timeout value invalid: Indefinite timeout enforced
Feb 16 08:48:04 localhost sshd[11882]: Failed password for radiant from 192.168.1.75 port 45886 ssh2

Feb 16 08:48:09 localhost sshd[11882]: Failed password for radiant from 192.168.1.75 port 45886 ssh2

Feb 16 08:48:12 localhost sshd[11882]: Failed password for radiant from 192.168.1.75 port 45886 ssh2

Blocking IP indefinitely
^CShutdown Application!
[root@localhost Assignment-2]#
```

- Before the test, iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

- During the test when attacker 1 exceeds maximum allowed attempts of 2 , iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  192.168.1.75         0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

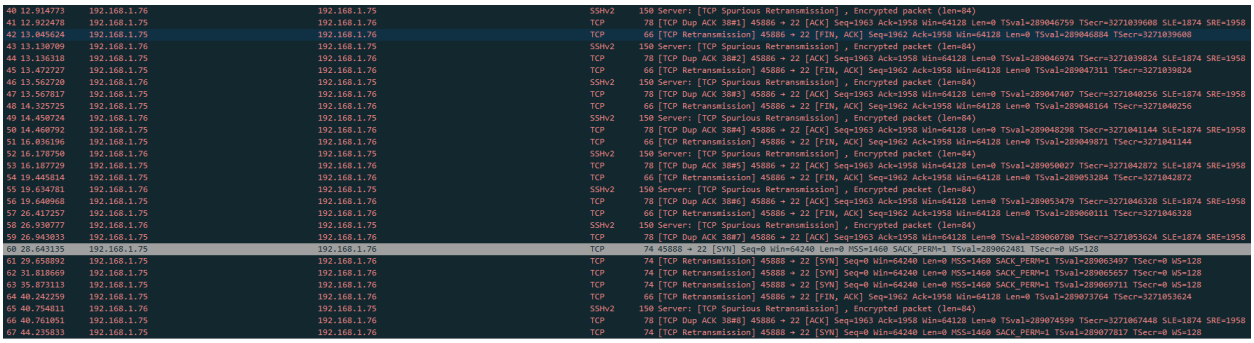
Iptables filters all inbound traffic to the server. This filter rule is not flushed until the program ends. This screenshot (along with the wireshark captures) confirm the working functionality to verify the test case.

- The log file /var/log/secure shows:

```
Feb 16 08:48:03 localhost sshd[11882]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 08:48:04 localhost sshd[11882]: Failed password for radiant from 192.168.1.75 port 45886 ssh2
Feb 16 08:48:09 localhost sshd[11882]: Failed password for radiant from 192.168.1.75 port 45886 ssh2
Feb 16 08:48:12 localhost sshd[11882]: Failed password for radiant from 192.168.1.75 port 45886 ssh2
Feb 16 08:49:08 localhost sshd[11882]: Connection closed by authenticating user radiant 192.168.1.75 port 45886 [preauth]
Feb 16 08:49:08 localhost sshd[11882]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
```

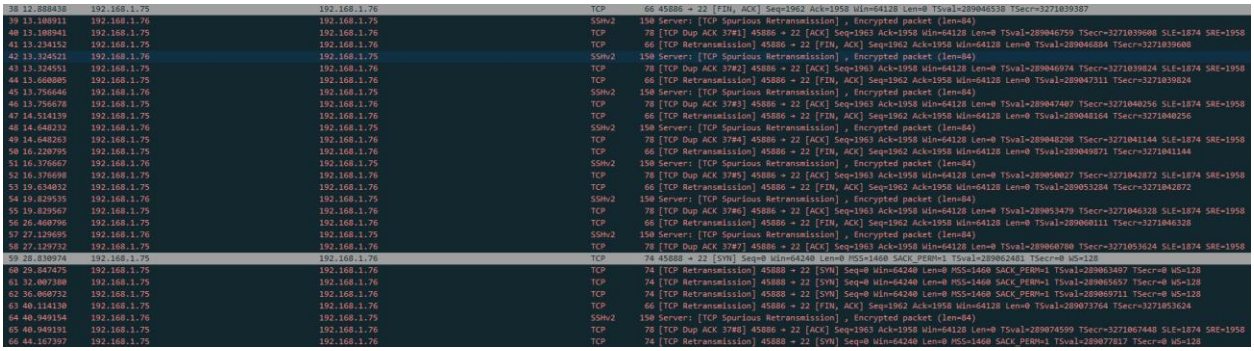
The attacking client may not at any point in time, after the rule is enforced, reconnect to the server.

- Server packet capture



40	12.914773	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
41	12.922478	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#1] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289046759 TSecr=3271039608 SLE=1874 SRE=1958
42	13.045624	192.168.1.76	192.168.1.75	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289046884 TSecr=3271039608
43	13.130709	192.168.1.75	192.168.1.76	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
44	13.136318	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#2] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289046974 TSecr=3271039824 SLE=1874 SRE=1958
45	13.472727	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289047311 TSecr=3271039824
46	13.562720	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
47	13.567817	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#3] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289047407 TSecr=3271040256 SLE=1874 SRE=1958
48	14.325725	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289048164 TSecr=3271040256
49	14.450724	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
50	14.460792	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#4] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289048298 TSecr=3271041144 SLE=1874 SRE=1958
51	16.036196	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289049871 TSecr=3271041144
52	16.178750	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
53	16.187729	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#5] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289050027 TSecr=3271042872 SLE=1874 SRE=1958
54	19.445814	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289053284 TSecr=3271042872
55	19.634781	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
56	19.640968	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#6] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289053479 TSecr=3271046328 SLE=1874 SRE=1958
57	26.417257	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289060111 TSecr=3271046328
58	26.930777	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
59	26.943033	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#7] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289060780 TSecr=3271053624 SLE=1874 SRE=1958
60	28.643135	192.168.1.75	192.168.1.76	TCP	74 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289062481 TSecr=0 WS=128
61	29.658892	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289063497 TSecr=0 WS=128
62	31.818669	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289065657 TSecr=0 WS=128
63	35.873113	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289069711 TSecr=0 WS=128
64	40.242259	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289073764 TSecr=3271053624
65	40.754811	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
66	40.761051	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 38#8] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289074599 TSecr=3271067448 SLE=1874 SRE=1958
67	44.235833	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289077817 TSecr=0 WS=128

- Attacker 1 packet capture



38	12.888438	192.168.1.75	192.168.1.76	TCP	66 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289046536 TSecr=3271039387
39	13.100911	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
40	13.108941	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#1] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289046759 TSecr=3271039608 SLE=1874 SRE=1958
41	13.234152	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289046884 TSecr=3271039608
42	13.324521	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
43	13.324551	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#2] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289046974 TSecr=3271039824 SLE=1874 SRE=1958
44	13.660805	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289047311 TSecr=3271039824
45	13.756646	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
46	13.756678	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#3] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289047407 TSecr=3271040256 SLE=1874 SRE=1958
47	14.514139	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289048164 TSecr=3271040256
48	14.648232	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
49	14.648263	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#4] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289048298 TSecr=3271041144 SLE=1874 SRE=1958
50	16.220795	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289049871 TSecr=3271041144
51	16.376667	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
52	16.376698	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#5] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289050027 TSecr=3271042872 SLE=1874 SRE=1958
53	19.634032	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289053284 TSecr=3271042872
54	19.829535	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
55	19.829567	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#6] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289053479 TSecr=3271046328 SLE=1874 SRE=1958
56	26.460796	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289060111 TSecr=3271046328
57	27.129695	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
58	27.129732	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#7] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289060780 TSecr=3271053624 SLE=1874 SRE=1958
59	28.830974	192.168.1.75	192.168.1.76	TCP	74 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289062481 TSecr=0 WS=128
60	29.847475	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289063497 TSecr=0 WS=128
61	32.007380	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289065657 TSecr=0 WS=128
62	36.060732	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289069711 TSecr=0 WS=128
63	40.114130	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45886 → 22 [FIN, ACK] Seq=1962 Ack=1958 Win=64128 Len=0 TSval=289073764 TSecr=3271053624
64	40.949154	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
65	40.949191	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#8] 45886 → 22 [ACK] Seq=1963 Ack=1958 Win=64128 Len=0 TSval=289074599 TSecr=3271067448 SLE=1874 SRE=1958
66	44.167397	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45888 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289077817 TSecr=0 WS=128

These packet captures show the retransmissions occurring from the timeout events happening due to the packets being dropped at the filter (lasting indefinitely)

Case 3

- For this test, the following arguments are given to the monitoring program along with the associated unauthorized attempts to connect to the server and measures taken. 2 maximum attempts to connect to the server were specified, and a timeout value of 15 seconds is enforced after attackers exceed the maximum attempts.

```
[root@localhost Assignment-2]# ./ssh-mon.py -a 2 -t 15
Number of allowed attempts for each ssh client before timeout: 2
Timeout of 15 seconds will be given after number of allowed attempts exceeded
Feb 16 08:54:06 localhost sshd[11921]: Failed password for radiant from 192.168.1.75 port 45918 ssh2

Feb 16 08:54:10 localhost sshd[11921]: Failed password for radiant from 192.168.1.75 port 45918 ssh2

Feb 16 08:54:17 localhost sshd[11921]: Failed password for radiant from 192.168.1.75 port 45918 ssh2

Blocking IP: 192.168.1.75 for 15 seconds
Blocking IP: 192.168.1.75       Time: 08:54:17
Unblocking IP: 192.168.1.75     Time: 08:54:32
Thread Unblocked at 08:54:32
```

- Before the test, iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

- During the test when attacker 1 exceeds maximum allowed attempts of 2 , iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  192.168.1.75         0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

- After the timeout value of 15 seconds, iptables shows:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

All rules are flushed, this confirms that our monitoring script was able to successfully remove the filter after the timeout completes. With the previous screenshots, this confirms the working functionality to verify the test case.

- The log file /var/log/secure shows:

```
Feb 16 08:54:04 localhost sshd[11921]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 08:54:06 localhost sshd[11921]: Failed password for radiant from 192.168.1.75 port 45918 ssh2
Feb 16 08:54:10 localhost sshd[11921]: Failed password for radiant from 192.168.1.75 port 45918 ssh2
Feb 16 08:54:17 localhost sshd[11921]: Failed password for radiant from 192.168.1.75 port 45918 ssh2
Feb 16 08:54:17 localhost sshd[11921]: Connection closed by authenticating user radiant 192.168.1.75 port 45918 [preauth]
Feb 16 08:54:17 localhost sshd[11921]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 08:54:42 localhost sshd[11930]: Accepted password for radiant from 192.168.1.75 port 45920 ssh2
Feb 16 08:54:42 localhost sshd[11930]: pam_unix(sshd:session): session opened for user radiant by (uid=0)
Feb 16 08:54:46 localhost sshd[11934]: Received disconnect from 192.168.1.75 port 45920:11: disconnected by user
Feb 16 08:54:46 localhost sshd[11934]: Disconnected from user radiant 192.168.1.75 port 45920
Feb 16 08:54:46 localhost sshd[11930]: pam_unix(sshd:session): session closed for user radiant
```

After the timeout, the attacking client may reconnect to the server

- Server packet capture

```
43 19.651154   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289415657 TSecr=0 WS=128
44 21.862616   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289417871 TSecr=0 WS=128
45 25.919158   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289421924 TSecr=0 WS=128
46 34.134101   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289430031 TSecr=0 WS=128
47 34.134269   192.168.1.76   192.168.1.75   TCP    74 22 → 45920 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=3271422997 TSecr=289430031 WS=128
48 34.151559   192.168.1.75   192.168.1.76   TCP    66 45920 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=289430148 TSecr=3271422997
49 34.151815   192.168.1.75   192.168.1.76   SSHv2  87 Client: Protocol (SSH-2.0-OpenSSH_8.4)
50 34.151871   192.168.1.76   192.168.1.75   TCP    66 22 → 45920 [ACK] Seq=1 Ack=22 Win=65152 Len=0 TSval=3271423015 TSecr=289430149
51 34.243925   192.168.1.76   192.168.1.75   SSHv2  87 Server: Protocol (SSH-2.0-OpenSSH_8.3)
```

- Attacker 1 packet capture

```
42 19.649028   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289415657 TSecr=0 WS=128
43 21.862268   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289417871 TSecr=0 WS=128
44 25.915655   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289421924 TSecr=0 WS=128
45 34.022365   192.168.1.75   192.168.1.76   TCP    74 [TCP Retransmission] 45920 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=289430031 TSecr=0 WS=128
46 34.139685   192.168.1.76   192.168.1.75   TCP    74 22 → 45920 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=3271422997 TSecr=289430031 WS=128
47 34.139746   192.168.1.75   192.168.1.76   TCP    66 45920 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=289430148 TSecr=3271422997
48 34.140475   192.168.1.75   192.168.1.76   SSHv2  87 Client: Protocol (SSH-2.0-OpenSSH_8.4)
49 34.157383   192.168.1.76   192.168.1.75   TCP    66 22 → 45920 [ACK] Seq=1 Ack=22 Win=65152 Len=0 TSval=3271423015 TSecr=289430149
50 34.254508   192.168.1.76   192.168.1.75   SSHv2  87 Server: Protocol (SSH-2.0-OpenSSH_8.3)
```

These packet captures show the retransmissions occurring from the timeout events happening due to the packets being dropped at the filter (lasting approximately 15 seconds). After the timeout is exceeded, the firewall rule is flushed allowing the attacking client to attempt connections again.

Case 4

- For this test, the following arguments are given to the monitoring program along with the associated unauthorized attempts to connect to the server and measures taken. 2 maximum attempts to connect to the server were specified, and a timeout value of 30 seconds is enforced after attackers exceed the maximum attempts. Two attacking clients attempt staggered access the SSH service.

```
Number of allowed attempts for each ssh client before timeout: 2
Timeout of 30 seconds will be given after number of allowed attempts exceeded
Feb 16 09:14:53 localhost sshd[12183]: Failed password for radiant from 192.168.1.75 port 45962 ssh2

Feb 16 09:15:00 localhost sshd[12183]: Failed password for radiant from 192.168.1.75 port 45962 ssh2

Feb 16 09:15:08 localhost sshd[12183]: Failed password for radiant from 192.168.1.75 port 45962 ssh2

Blocking IP: 192.168.1.75 for 30 seconds
Blocking IP: 192.168.1.75       Time: 09:15:09
Unblocking IP: 192.168.1.75     Time: 09:15:39
Feb 16 09:15:12 localhost sshd[12185]: Failed password for radiant from 192.168.1.78 port 48698 ssh2

Feb 16 09:15:18 localhost sshd[12185]: Failed password for radiant from 192.168.1.78 port 48698 ssh2

Feb 16 09:15:24 localhost sshd[12185]: Failed password for radiant from 192.168.1.78 port 48698 ssh2

Blocking IP: 192.168.1.78 for 30 seconds
Blocking IP: 192.168.1.78       Time: 09:15:25
Unblocking IP: 192.168.1.78     Time: 09:15:55
Thread Unblocked at 09:15:39
Thread Unblocked at 09:15:55
```

- Before the test, iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

- During the test when attacker 1 and 2 exceeds maximum allowed attempts of 2 , iptables show:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  192.168.1.75         0.0.0.0/0
DROP       all  --  192.168.1.78         0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

This confirms the working functionality to verify the test case.

- After the timeout value of 15 seconds, iptables shows:

```
[root@localhost radiant]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@localhost radiant]#
```

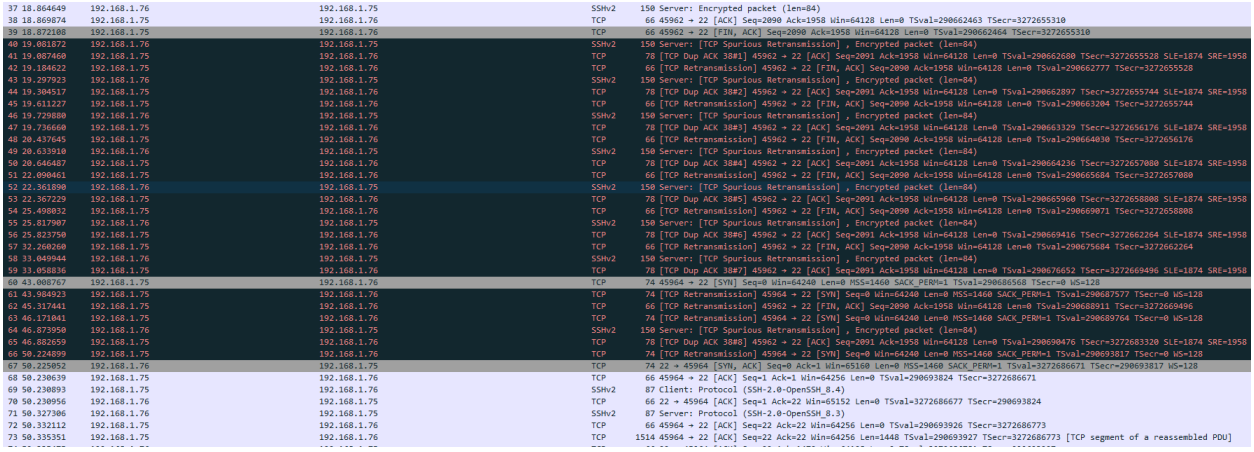
All rules are flushed, this confirms that our monitoring script was able to successfully remove the filter after the timeout completes.

- The log file /var/log/secure shows:

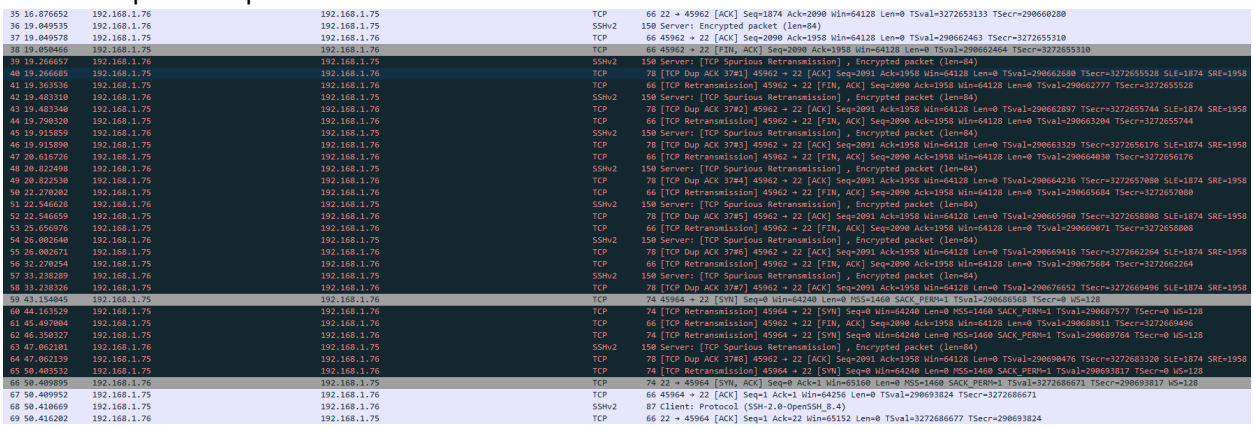
```
Feb 16 09:14:52 localhost sshd[12183]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 09:14:53 localhost sshd[12183]: Failed password for radiant from 192.168.1.75 port 45962 ssh2
Feb 16 09:15:00 localhost sshd[12183]: Failed password for radiant from 192.168.1.75 port 45962 ssh2
Feb 16 09:15:08 localhost sshd[12183]: Failed password for radiant from 192.168.1.75 port 45962 ssh2
Feb 16 09:15:10 localhost sshd[12185]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.78  user=radiant
Feb 16 09:15:12 localhost sshd[12185]: Failed password for radiant from 192.168.1.78 port 48698 ssh2
Feb 16 09:15:18 localhost sshd[12185]: Failed password for radiant from 192.168.1.78 port 48698 ssh2
Feb 16 09:15:24 localhost sshd[12185]: Failed password for radiant from 192.168.1.78 port 48698 ssh2
Feb 16 09:15:25 localhost sshd[12185]: Connection closed by authenticating user radiant 192.168.1.78 port 48698 [preauth]
Feb 16 09:15:25 localhost sshd[12185]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.78  user=radiant
Feb 16 09:15:47 localhost sshd[12196]: Accepted password for radiant from 192.168.1.75 port 45964 ssh2
Feb 16 09:15:47 localhost sshd[12196]: pam_unix(sshd:session): session opened for user radiant by (uid=0)
Feb 16 09:16:03 localhost sshd[12183]: Connection closed by authenticating user radiant 192.168.1.75 port 45962 [preauth]
Feb 16 09:16:03 localhost sshd[12183]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.75  user=radiant
Feb 16 09:16:06 localhost sshd[12230]: Accepted password for radiant from 192.168.1.78 port 48702 ssh2
Feb 16 09:16:06 localhost sshd[12230]: pam_unix(sshd:session): session opened for user radiant by (uid=0)
```

After the timeout, the attacking client may reconnect to the server

- Server packet capture

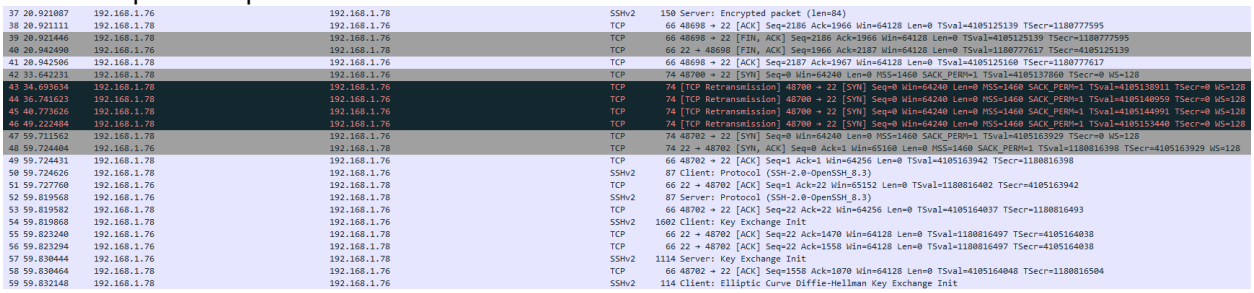


- Attacker 1 packet capture



35	16.876652	192.168.1.76	192.168.1.75	TCP	66 22 → 45962 [ACK] Seq=1874 Ack=2090 Win=64128 Len=0 TSval=3272653133 TSecr=290660280
36	19.049535	192.168.1.76	192.168.1.75	SSHv2	150 Server: Encrypted packet (len=84)
37	19.049578	192.168.1.75	192.168.1.76	TCP	66 45962 → 22 [ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290662463 TSecr=3272655310
38	19.050466	192.168.1.76	192.168.1.76	TCP	66 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290662464 TSecr=3272655310
39	19.266657	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
40	19.266685	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#1] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290662680 TSecr=3272655528 SLE=1874 SRE=1958
41	19.363536	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290662777 TSecr=3272655528
42	19.483310	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
43	19.483340	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#2] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290662897 TSecr=3272655744 SLE=1874 SRE=1958
44	19.790320	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290663204 TSecr=3272655744
45	19.915850	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
46	19.915890	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#3] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290663329 TSecr=3272656176 SLE=1874 SRE=1958
47	20.616726	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290664030 TSecr=3272656176
48	20.822498	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
49	20.822530	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#4] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290664236 TSecr=3272657080 SLE=1874 SRE=1958
50	22.270202	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290665684 TSecr=3272657080
51	22.546628	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
52	22.546659	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#5] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290665960 TSecr=3272658808 SLE=1874 SRE=1958
53	25.656976	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290669071 TSecr=3272658808
54	26.002640	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
55	26.002671	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#6] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290669416 TSecr=3272662264 SLE=1874 SRE=1958
56	32.270254	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290675684 TSecr=3272662264
57	33.238289	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
58	33.238326	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#7] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290676652 TSecr=3272669496 SLE=1874 SRE=1958
59	43.154045	192.168.1.75	192.168.1.76	TCP	74 45964 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=290686568 TSecr=0 WS=128
60	44.163529	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45964 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=290687577 TSecr=0 WS=128
61	45.497004	192.168.1.75	192.168.1.76	TCP	66 [TCP Retransmission] 45962 → 22 [FIN, ACK] Seq=2090 Ack=1958 Win=64128 Len=0 TSval=290688911 TSecr=3272669496
62	46.350327	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45964 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=290689764 TSecr=0 WS=128
63	47.062101	192.168.1.76	192.168.1.75	SSHv2	150 Server: [TCP Spurious Retransmission] , Encrypted packet (len=84)
64	47.062139	192.168.1.75	192.168.1.76	TCP	78 [TCP Dup ACK 37#8] 45962 → 22 [ACK] Seq=2091 Ack=1958 Win=64128 Len=0 TSval=290690476 TSecr=3272683320 SLE=1874 SRE=1958
65	50.403532	192.168.1.75	192.168.1.76	TCP	74 [TCP Retransmission] 45964 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=290693817 TSecr=0 WS=128
66	50.409895	192.168.1.76	192.168.1.75	TCP	74 22 → 45964 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=3272686671 TSecr=290693817 WS=128
67	50.409952	192.168.1.75	192.168.1.76	TCP	66 45964 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=290693824 TSecr=3272686671
68	50.410669	192.168.1.75	192.168.1.76	SSHv2	87 Client: Protocol (SSH-2.0-OpenSSH_8.4)
69	50.416202	192.168.1.76	192.168.1.75	TCP	66 22 → 45964 [ACK] Seq=1 Ack=22 Win=65152 Len=0 TSval=3272686677 TSecr=290693824

- Attacker 2 packet capture



37	20.921087	192.168.1.76	192.168.1.78	SSHv2	150 Server: Encrypted packet (len=84)
38	20.921111	192.168.1.78	192.168.1.76	TCP	66 48698 → 22 [ACK] Seq=2186 Ack=1966 Win=64128 Len=0 TSval=4105125139 TSecr=1180777595
39	20.921446	192.168.1.78	192.168.1.76	TCP	66 48698 → 22 [FIN, ACK] Seq=2186 Ack=1966 Win=64128 Len=0 TSval=4105125139 TSecr=1180777595
40	20.942490	192.168.1.76	192.168.1.78	TCP	66 22 → 48698 [FIN, ACK] Seq=1966 Ack=2187 Win=64128 Len=0 TSval=1180777617 TSecr=4105125139
41	20.942506	192.168.1.78	192.168.1.76	TCP	66 48698 → 22 [ACK] Seq=2187 Ack=1967 Win=64128 Len=0 TSval=4105125160 TSecr=1180777617
42	33.642231	192.168.1.78	192.168.1.76	TCP	74 48700 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4105137860 TSecr=0 WS=128
43	34.693634	192.168.1.78	192.168.1.76	TCP	74 [TCP Retransmission] 48700 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4105138911 TSecr=0 WS=128
44	36.741623	192.168.1.78	192.168.1.76	TCP	74 [TCP Retransmission] 48700 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4105140959 TSecr=0 WS=128
45	40.773626	192.168.1.78	192.168.1.76	TCP	74 [TCP Retransmission] 48700 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4105144991 TSecr=0 WS=128
46	49.222484	192.168.1.78	192.168.1.76	TCP	74 [TCP Retransmission] 48700 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4105153440 TSecr=0 WS=128
47	59.711562	192.168.1.78	192.168.1.76	TCP	74 48702 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4105163929 TSecr=0 WS=128
48	59.724404	192.168.1.76	192.168.1.78	TCP	74 22 → 48702 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=1180816398 TSecr=4105163929 WS=128
49	59.724431	192.168.1.78	192.168.1.76	TCP	66 48702 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4105163942 TSecr=1180816398
50	59.724626	192.168.1.78	192.168.1.76	SSHv2	87 Client: Protocol (SSH-2.0-OpenSSH_8.3)
51	59.727760	192.168.1.78	192.168.1.78	TCP	66 22 → 48702 [ACK] Seq=1 Ack=22 Win=65152 Len=0 TSval=1180816402 TSecr=4105163942
52	59.819568	192.168.1.78	192.168.1.78	SSHv2	87 Server: Protocol (SSH-2.0-OpenSSH_8.3)
53	59.819582	192.168.1.78	192.168.1.76	TCP	66 48702 → 22 [ACK] Seq=22 Ack=22 Win=64256 Len=0 TSval=4105164037 TSecr=1180816493
54	59.819868	192.168.1.78	192.168.1.76	SSHv2	1602 Client: Key Exchange Init
55	59.823240	192.168.1.76	192.168.1.78	TCP	66 22 → 48702 [ACK] Seq=22 Ack=1470 Win=64128 Len=0 TSval=1180816497 TSecr=4105164038
56	59.823294	192.168.1.76	192.168.1.78	TCP	66 22 → 48702 [ACK] Seq=22 Ack=1558 Win=64128 Len=0 TSval=1180816497 TSecr=4105164038
57	59.830444	192.168.1.76	192.168.1.78	SSHv2	1114 Server: Key Exchange Init
58	59.830464	192.168.1.78	192.168.1.76	TCP	66 48702 → 22 [ACK] Seq=1558 Ack=1070 Win=64128 Len=0 TSval=4105164048 TSecr=1180816504
59	59.832148	192.168.1.78	192.168.1.76	SSHv2	114 Client: Elliptic Curve Diffie-Hellman Key Exchange Init

These packet captures show the retransmissions occurring from the timeout events happening due to the packets being dropped at the filter (lasting approximately 30 seconds). After the timeout is exceeded, the firewall rule is flushed allowing the attacking clients to attempt connections again. Note, the differences in the number of TCP Retransmissions can be accounted for due to the nature of the test environment. Attacker 1 was maintaining a SSH connection through a PuTTY client throughout the test whereas Attacker 2 was not.