Firewall

Lab Report 5

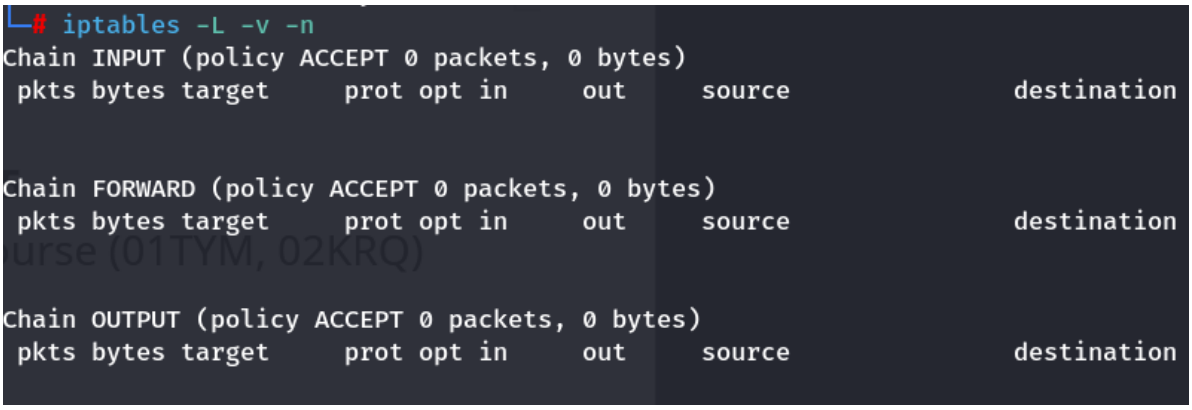
Information Systems Security course (01TYM, 02KRQ)

prepared by:

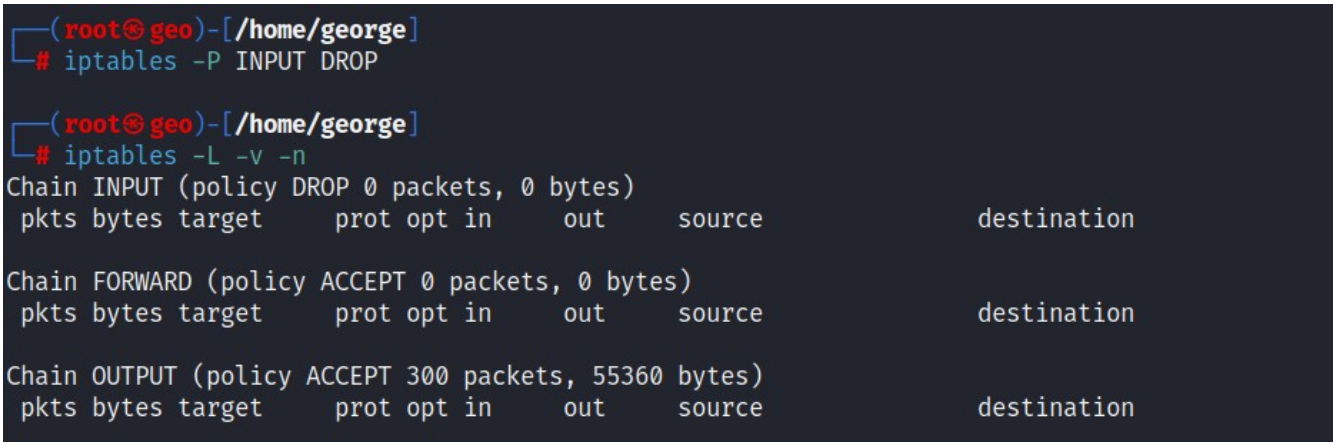
Anuar Elio Magliari (s317033@studenti.polito.it)
George Florin Eftime (s303483@studenti.polito.it)
Alekos Interrante Bonadia (s319849@studenti.polito.it)

Packet filter

- Which authorisation policy is configured by default on your machine (on each of the three chains)?



- Which chain (out from INPUT, FORWARD, and OUTPUT) do you have to modify to protect your machine from connections originating from the external users?
  - INPUT and FORWARD in order to have protection from external users



- Write down the iptables command to modify the authorisation policy of Alice's host, so that to reject any traffic (hint: you need to modify the default policy for the INPUT chain from ACCEPT to DROP):

- Does Bob receive any responses (to the ping) from Alice's host?
  - No, he doesn't

```
┌──(elion-man㉿Elion-Man-on-Kali)-[~]
└─$ ping 172.22.17.139
PING 172.22.17.139 (172.22.17.139) 56(84) bytes of data.
^C
── 172.22.17.139 ping statistics ──
31 packets transmitted, 0 received, 100% packet loss, time 30701ms
```

- Can Bob connect to Alice's host via SSH and HTTP (with the browser)?
  - No, he doesn't. In both cases, Bob doesn't receive any response from Alice because Alice's firewall has the INPUT chain set to DROP, so it discards all requests from the external source.
- Check with nmap (running the above indicated nmap command on Bob's host) the status of the ports 22 and 80 on Alice's host. What is their status now?

```
└─# nmap -sT -Pn -n -p 80,22 -v 172.22.17.139
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 12:24 CET
Initiating Connect Scan at 12:24
Scanning 172.22.17.139 [2 ports]
Completed Connect Scan at 12:24, 3.00s elapsed (2 total ports)
Nmap scan report for 172.22.17.139
Host is up.

PORT   STATE    SERVICE
22/tcp filtered ssh
80/tcp filtered http

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 3.04 seconds
```

- Write down the iptables command to add a rule to the authorisation policy on Alice's host (for the input traffic), so that to enable all ICMP traffic (for simplicity, we provide you some of the parameters of the command)

- iptables -A INPUT -p icmp -j ACCEPT

```
┌──(root㉿geo)-[/var/www/html]
└─# iptables -L -v -n
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    6   504 ACCEPT     1    --  *      *       172.22.16.108        0.0.0.0/0

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

Chain OUTPUT (policy ACCEPT 1292 packets, 547K bytes)
 pkts bytes target     prot opt in     out     source               destination
```

- Does Bob receive this time any response from Alice's host in response to the ping command?
  - Yes, now he does because Alice allows all the icmp packets in input.

```
┌──(elion-man㉿Elion-Man-on-Kali)-[~]
└─$ ping 172.22.17.139
PING 172.22.17.139 (172.22.17.139) 56(84) bytes of data.
64 bytes from 172.22.17.139: icmp_seq=1 ttl=64 time=7.40 ms
64 bytes from 172.22.17.139: icmp_seq=2 ttl=64 time=7.32 ms
64 bytes from 172.22.17.139: icmp_seq=3 ttl=64 time=5.06 ms
64 bytes from 172.22.17.139: icmp_seq=4 ttl=64 time=4.27 ms
64 bytes from 172.22.17.139: icmp_seq=5 ttl=64 time=17.3 ms
64 bytes from 172.22.17.139: icmp_seq=6 ttl=64 time=19.5 ms
64 bytes from 172.22.17.139: icmp_seq=7 ttl=64 time=30.6 ms
64 bytes from 172.22.17.139: icmp_seq=8 ttl=64 time=7.02 ms
^C
── 172.22.17.139 ping statistics ──
8 packets transmitted, 8 received, 0% packet loss, time 7013ms
rtt min/avg/max/mdev = 4.271/12.311/30.586/8.699 ms
```

- Next, on Alice's host, write down the iptables command to allow the TCP input traffic towards the port 80 (on Alice):
  - iptables -A INPUT -p tcp --dport 80 -j ACCEPT

- Check out the configuration of IPtables on Alice's host. Which chain has been modified?
  - The chain modified is the INPUT chain as specified in the command

```
┌──(root㉿geo)-[/var/www/html]
└─# ./script_firewall.sh
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     1    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     6    --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:80

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

Chain OUTPUT (policy ACCEPT 1544 packets, 609K bytes)
 pkts bytes target     prot opt in     out     source               destination
```

- Does Chuck receive any response (to the ping) from the Alice's host? Why?
  - Yes, Chuck receives ping responses from Alice because she doesn't drop the icmp requests (source is set to anyone)

```
┌──(kali㉿kali)-[~]
└─$ nmap -sT -Pn -n -p 80,22 -v 172.22.17.186
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-05 15:35 UTC
Initiating Connect Scan at 15:35
Scanning 172.22.17.186 [2 ports]
Discovered open port 80/tcp on 172.22.17.186
Completed Connect Scan at 15:35, 1.61s elapsed (2 total ports)
Nmap scan report for 172.22.17.186
Host is up (0.061s latency).

PORT   STATE    SERVICE
22/tcp filtered ssh
80/tcp open     http

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 1.66 seconds
```

- How can nmap distinguish between filtered ports and closed ports? Verify by analysing the traffic exchanged between the two machines (e.g. with wireshark).
  - Nmap distinguishes between these two states based on whether it receives a response to its probe. In the case of a closed port, it receives a TCP RST packet in response, whereas for a filtered port, it doesn't receive any packet.
- What is the status of the port 80 (as listed by the above nmap command)?
  - The status of the port 80 is set to filtered.

```
└─$ nmap -sT -Pn -n -p 80 -v 172.22.17.170
Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-12-12 09:41 CET
Initiating Connect Scan at 09:41
Scanning 172.22.17.170 [1 port]
Completed Connect Scan at 09:41, 2.00s elapsed (1 total ports)
Nmap scan report for 172.22.17.170
Host is up.

PORT   STATE    SERVICE
80/tcp filtered http

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 2.04 seconds
```

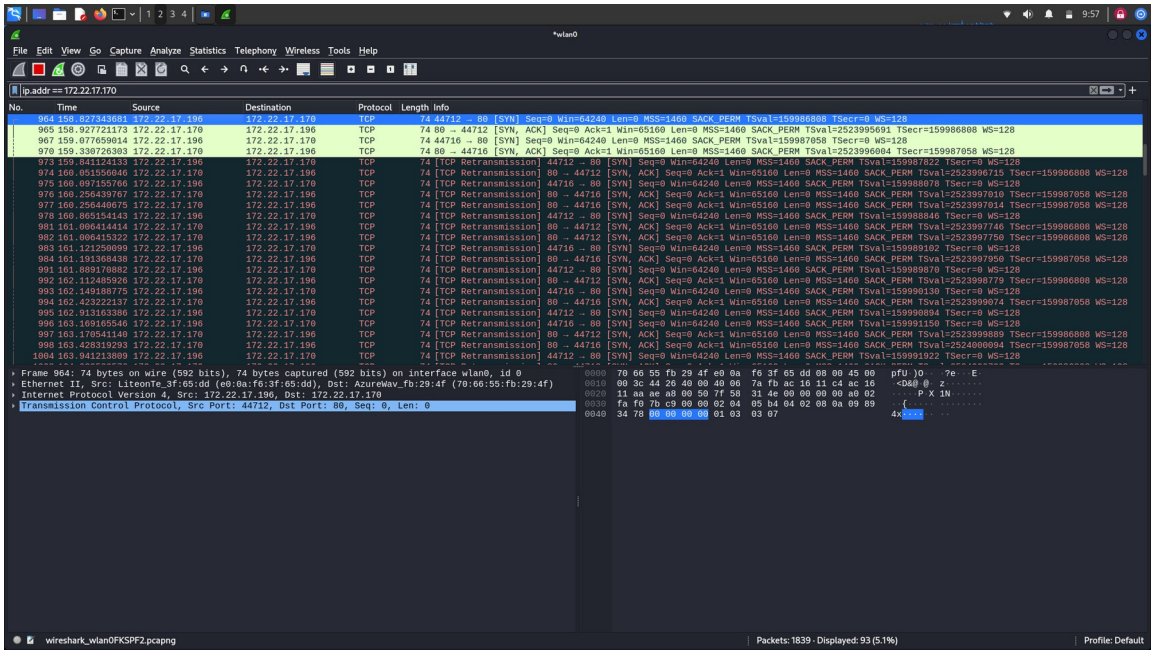
- What is now the status of the port 80 on Bob (as listed by the above nmap command on Alice's host)?
  - The status of the port 80 is still set to filtered.

```
└─$ nmap -sT -Pn -n -p 80 -v 172.22.17.170
Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-12-12 09:42 CET
Initiating Connect Scan at 09:42
Scanning 172.22.17.170 [1 port]
Completed Connect Scan at 09:42, 2.00s elapsed (1 total ports)
Nmap scan report for 172.22.17.170
Host is up.

PORT   STATE    SERVICE
80/tcp filtered http

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 2.04 seconds
```

- Can Alice browse web content on Bob's host? Verify by analysing the traffic among the two machines (with wireshark). After analyzing the traffic with Wireshark, describe what is happening with the SYN, SYN ACK, and ACK messages exchanged between Alice and Bob when Alice tries to browse web content on Bob's host.
  - Alice is unable to browse the web content on Bob's host because she sends a SYN packet to Bob, and Bob responds with a SYN-ACK packet. However, Alice drops the response because the policies specify allowing packets from the destination port equal to 80, whereas it is not allowed to accept packets with the source port equal to 80. Consequently, Alice retransmits the SYN packet because, from her perspective, she doesn't receive the response (SYN-ACK). When the timeout expires, she retries to send the SYN packet.



- Before passing to the execution of the next exercise, restore on Alice's host the authorisation policy of type "ACCEPT ALL". Hint: You need to delete all the current rules and specify afterwards the rules for the input chain.
  - sudo iptables -F INPUT
  - sudo iptables -P INPUT ACCEPT

```
┌──(elion-man㉿Elion-Man-on-Kali)-[~/Desktop/LAB_IIS/LAB_5]
└─$ sudo iptables -L -v -n

Chain INPUT (policy ACCEPT 996 packets, 619K bytes)
 pkts bytes target     prot opt in     out     source               destination

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

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