LAB07 PoD simulation

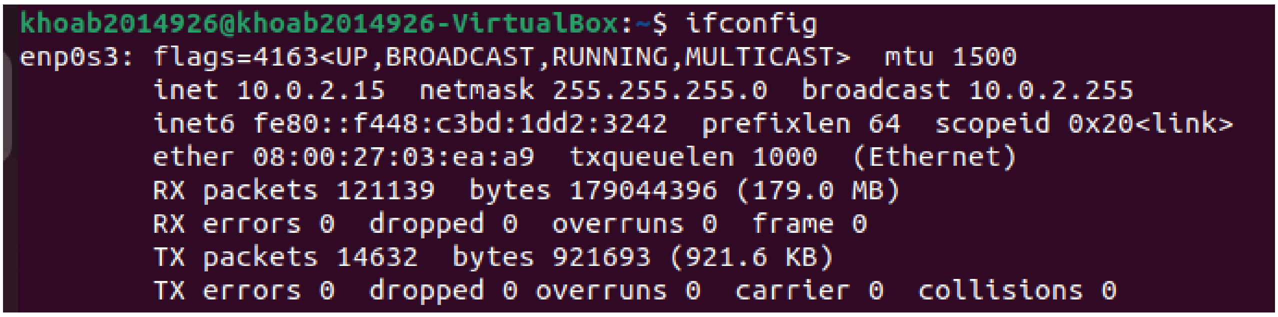
Class	M02
Student ID	B201926
Name	Trần Đăng Khoa
Email address	Khoab2014926@student.ctu.edu.vn
Browser	Safari, Chrome, IE, Firefox

1. Design test environment

	Attacker	Target
OS	Windows 10	Ubuntu
IP address	Test - bed IP	Test - bed IP
Attacking type	Ping of Death with 65000bytes	
Program for attacking	**Powershell, CMD**	
Command for monitoring		Gnome, netstat

[Powershell]

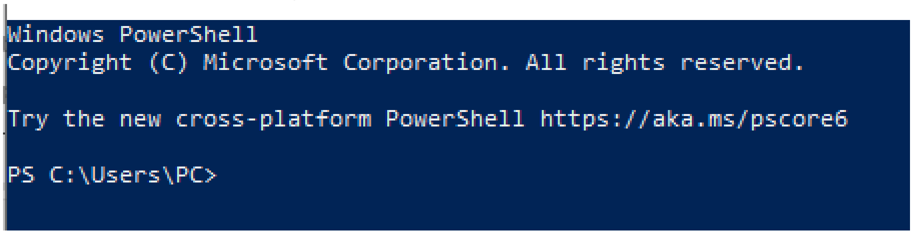
Taget:



```
khoab2014926@khoab2014926-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::f448:c3bd:1dd2:3242  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:03:ea:a9  txqueuelen 1000  (Ethernet)
        RX packets 121139  bytes 179044396 (179.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 14632  bytes 921693 (921.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

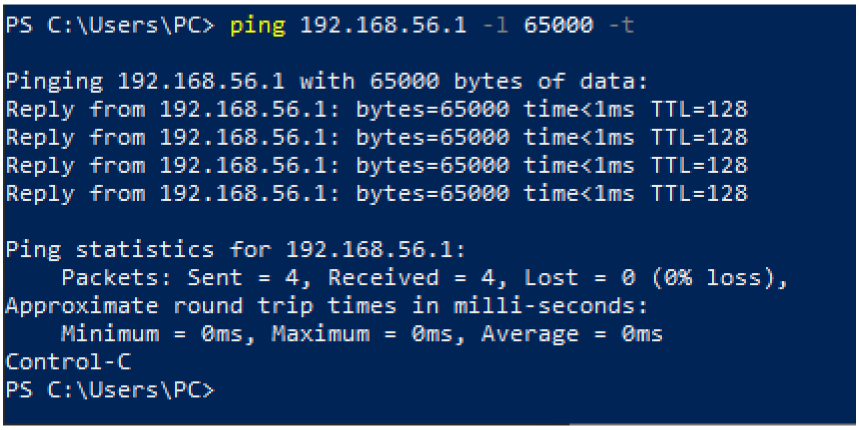
On attacker:

① Install **Powershell on attacker system**



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\PC>
```

② Send 65000-byte packets 5 times to ubuntu server using CMD ping command, **Powershell**:



```
PS C:\Users\PC> ping 192.168.56.1 -l 65000 -t

Pinging 192.168.56.1 with 65000 bytes of data:
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128

Ping statistics for 192.168.56.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
PS C:\Users\PC>
```

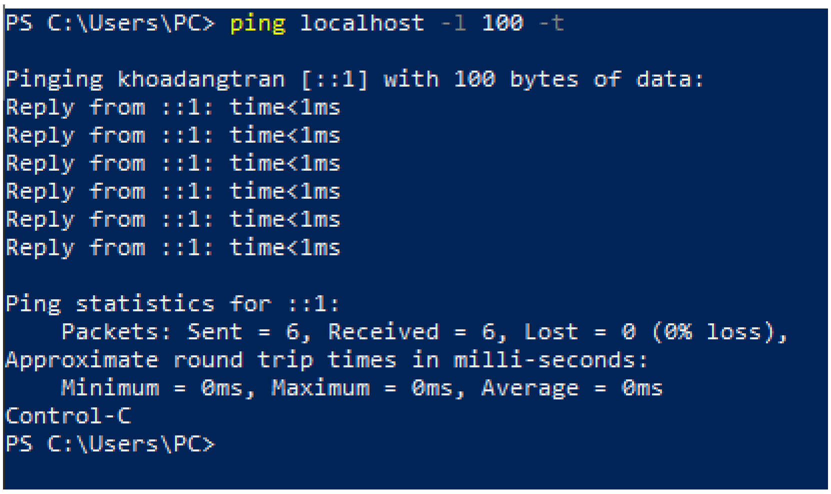
Option: **Powershell**

- -t means the data packets should be sent until the program is stopped
- -l specifies the data load to be sent to the victim
    - every few seconds this network receives about ~50kb, and it does not respond (sending – red line), because the packet size is exceeded, it cannot be processed

[PS to localhost]

PS C:₩Users₩Happy> ping localhost -l 100 -t

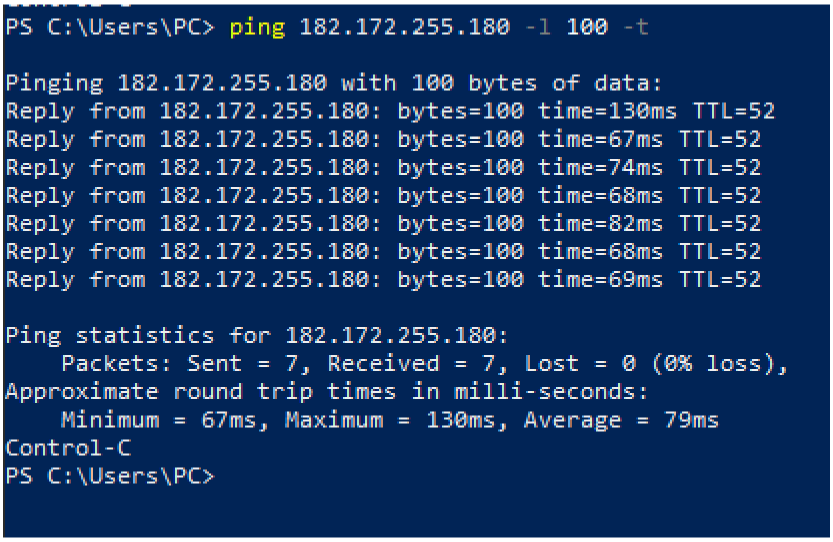
Control-C



```
PS C:\Users\PC> ping localhost -l 100 -t

Pinging khoadangtran [::1] with 100 bytes of data:
Reply from ::1: time<1ms
Reply from ::1: time<1ms
Reply from ::1: time<1ms
Reply from ::1: time<1ms
Reply from ::1: time<1ms
Reply from ::1: time<1ms

Ping statistics for ::1:
    Packets: Sent = 6, Received = 6, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
PS C:\Users\PC>
```

[PS to CTU] PS C:\Users\Happy> ping 182.172.255.180 -l 100 -t

Control-C



```
PS C:\Users\PC> ping 182.172.255.180 -l 100 -t

Pinging 182.172.255.180 with 100 bytes of data:
Reply from 182.172.255.180: bytes=100 time=130ms TTL=52
Reply from 182.172.255.180: bytes=100 time=67ms TTL=52
Reply from 182.172.255.180: bytes=100 time=74ms TTL=52
Reply from 182.172.255.180: bytes=100 time=68ms TTL=52
Reply from 182.172.255.180: bytes=100 time=82ms TTL=52
Reply from 182.172.255.180: bytes=100 time=68ms TTL=52
Reply from 182.172.255.180: bytes=100 time=69ms TTL=52

Ping statistics for 182.172.255.180:
    Packets: Sent = 7, Received = 7, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 67ms, Maximum = 130ms, Average = 79ms
Control-C
PS C:\Users\PC>
```

CMD

Option: CMD

- -t means the data packets should be sent until the program is stopped •

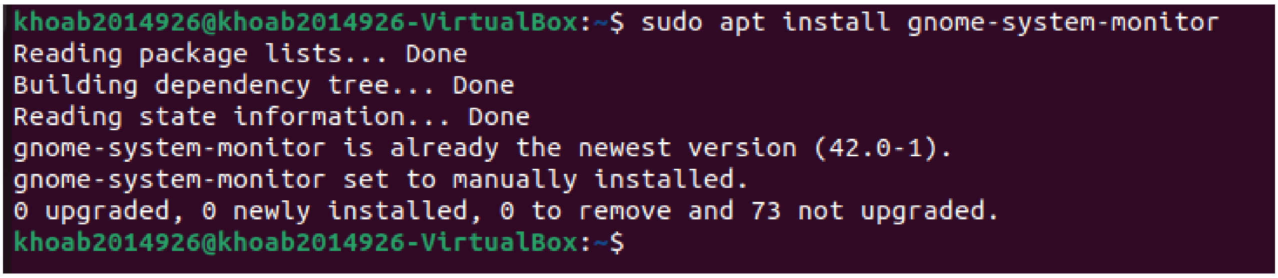
  -l specifies the data load to be sent to the victim

On target:

every few seconds this network receives about ~50kb, and it does not respond (sending – red line), because the packet size is exceeded, it cannot be processed

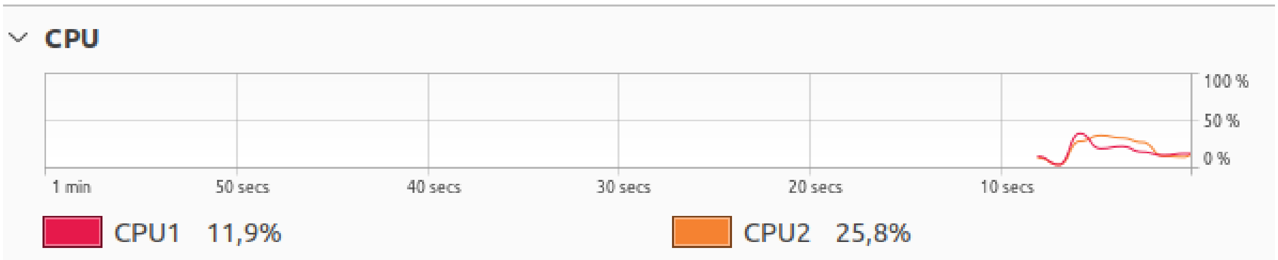
③ Install Linux monitoring software Gnome on target and analyze target system

Explain: To install GNOME System Monitor on Ubunt

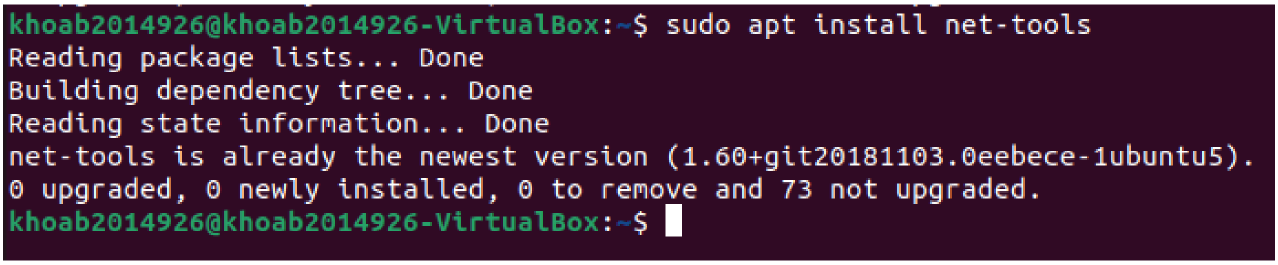


```
khoab2014926@khoab2014926-VirtualBox:~$ sudo apt install gnome-system-monitor
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gnome-system-monitor is already the newest version (42.0-1).
gnome-system-monitor set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 73 not upgraded.
khoab2014926@khoab2014926-VirtualBox:~$
```

Check the traffic volume of target system using relevant tool

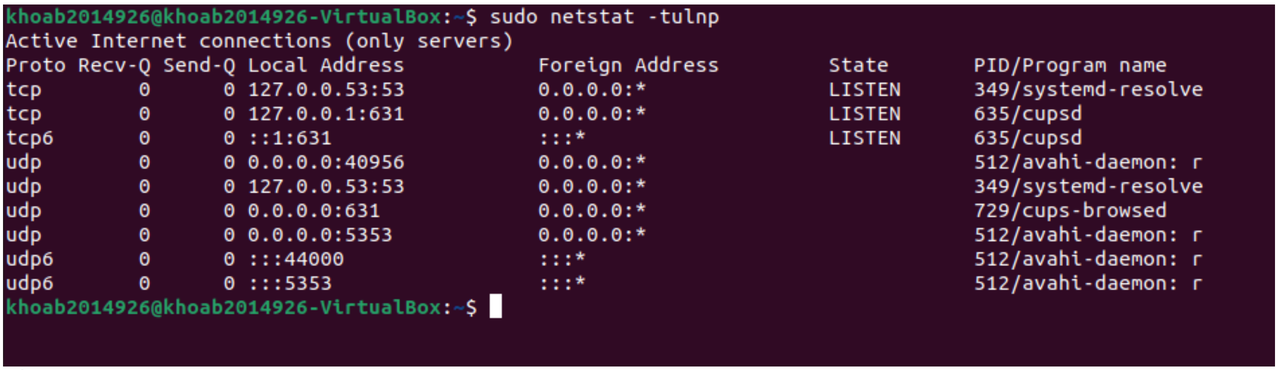


④ Install CMD net-tools package on Ubuntu and explain attacking result



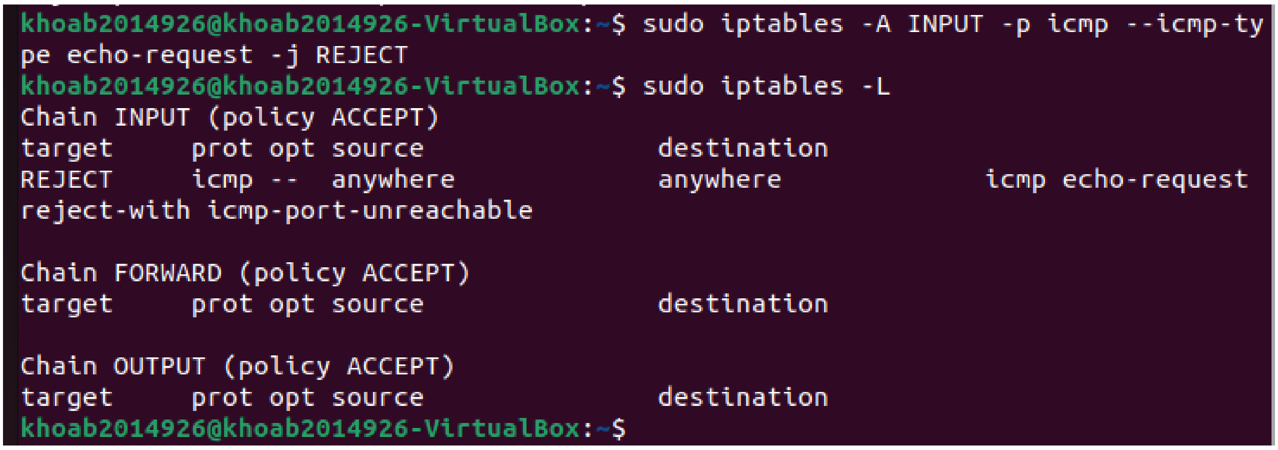
```
khoab2014926@khoab2014926-VirtualBox:~$ sudo apt install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
net-tools is already the newest version (1.60+git20181103.0eebece-1ubuntu5).
0 upgraded, 0 newly installed, 0 to remove and 73 not upgraded.
khoab2014926@khoab2014926-VirtualBox:~$ 
```

⑤ Detect DoS attack Symptom on the target system with CMD netstat commands

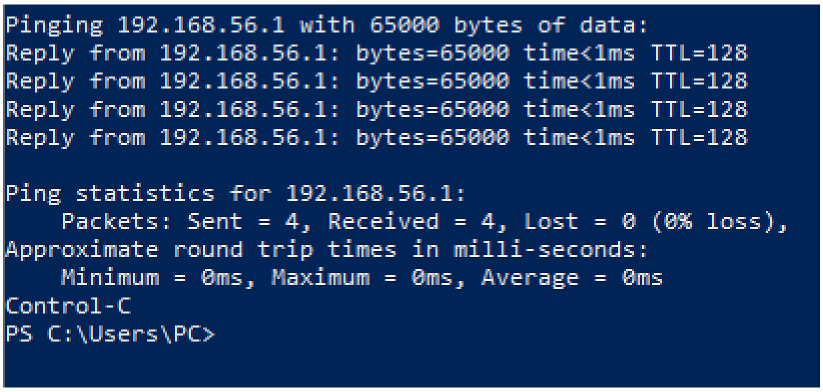


```
khoab2014926@khoab2014926-VirtualBox:~$ sudo netstat -tulnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      349/systemd-resolve
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      635/cupsd
tcp6       0      0 ::1:631                 :::*                    LISTEN      635/cupsd
udp        0      0 0.0.0.0:40956           0.0.0.0:*                           512/avahi-daemon: r
udp        0      0 127.0.0.53:53           0.0.0.0:*                           349/systemd-resolve
udp        0      0 0.0.0.0:631             0.0.0.0:*                           729/cups-browsed
udp        0      0 0.0.0.0:5353            0.0.0.0:*                           512/avahi-daemon: r
udp6       0      0 :::44000                :::*                                512/avahi-daemon: r
udp6       0      0 :::5353                 :::*                                512/avahi-daemon: r
khoab2014926@khoab2014926-VirtualBox:~$ 
```

⑥ Block DoS attack IP on Ubuntu using commands iptables (snap shot) and explain blocking result



```
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -A INPUT -p icmp --icmp-ty
pe echo-request -j REJECT
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination         
REJECT     icmp --  anywhere             anywhere             icmp echo-request
reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination         

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination         
khoab2014926@khoab2014926-VirtualBox:~$
```



```
Pinging 192.168.56.1 with 65000 bytes of data:
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128
Reply from 192.168.56.1: bytes=65000 time<1ms TTL=128

Ping statistics for 192.168.56.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
PS C:\Users\PC>
```

Explain:

Pinging 192.168.56.1 with 65000 bytes of data: Send ICMP echo requests (ping) to the IP address "192.168.56.1"

Reply from 192.168.56.1: Destination port unreachable: The server at IP address "192.168.56.1" received your ICMP echo requests but responded with "Destination port unreachable" messages.

- ICMP is a network protocol used for various network-related messages, including "ping."
- The "Destination port unreachable" message typically indicates that the destination (the server) received the packet but couldn't find a listening service or port to which it should be delivered.

Ping Statistics: The statistics at the end of the output provide information about the ping operation:

- "Packets: Sent = 4": Sent 4 ICMP echo requests.
- "Received = 4": Received 4 responses.
- "Lost = 0 (0% loss)": None of the packets were lost in transit, indicating that all 4 ICMP echo requests reached the destination and received responses.

⑦ Explain how to Block/Allow ping from iptables?

Block Ping

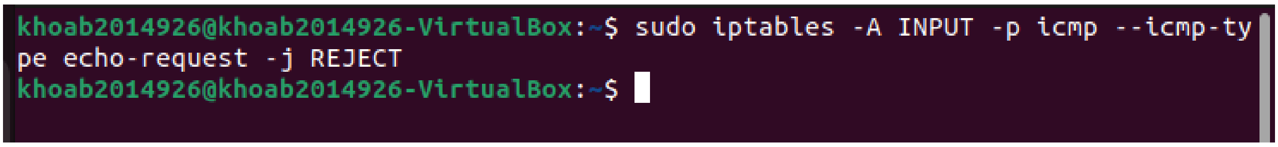
$sudo iptables -A INPUT -p icmp --icmp-type echo-request -j REJECT

Explain: the command sudo iptables -A INPUT -p icmp --icmp-type echo-request -j REJECT adds a rule to the INPUT chain of iptables that rejects all incoming ICMP echo requests. This means that no other hosts on the network will be able to ping your computer. -A INPUT: This appends the rule to the INPUT chain, which handles incoming packets.

-p icmp: It specifies that the rule applies to the ICMP protocol.

--icmp-type echo-request: This further specifies the rule to match ICMP echo requests (ping requests).

-j REJECT: This means that any incoming ICMP echo request packets will be rejected and not allowed to reach their destination. The sender of the ICMP echo request will receive an ICMP "Destination Unreachable" message in response.



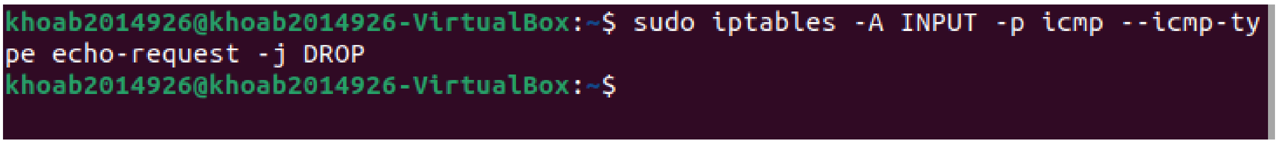
```
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -A INPUT -p icmp --icmp-ty
pe echo-request -j REJECT
khoab2014926@khoab2014926-VirtualBox:~$ 
```

Or else, you can add the following rules in order to block ping without printing an error message:

$sudo iptables -A INPUT -p icmp --icmp-type echo-request -j DROP

Explain: the command sudo iptables -A INPUT -p icmp --icmp-type echo-request -j DROP adds a rule to the INPUT chain of iptables that drops all incoming ICMP echo requests. This means that no other hosts on the network will be able to ping your computer.

-j DROP: This part of the command instructs the firewall to drop (discard) any incoming ICMP echo request packets.

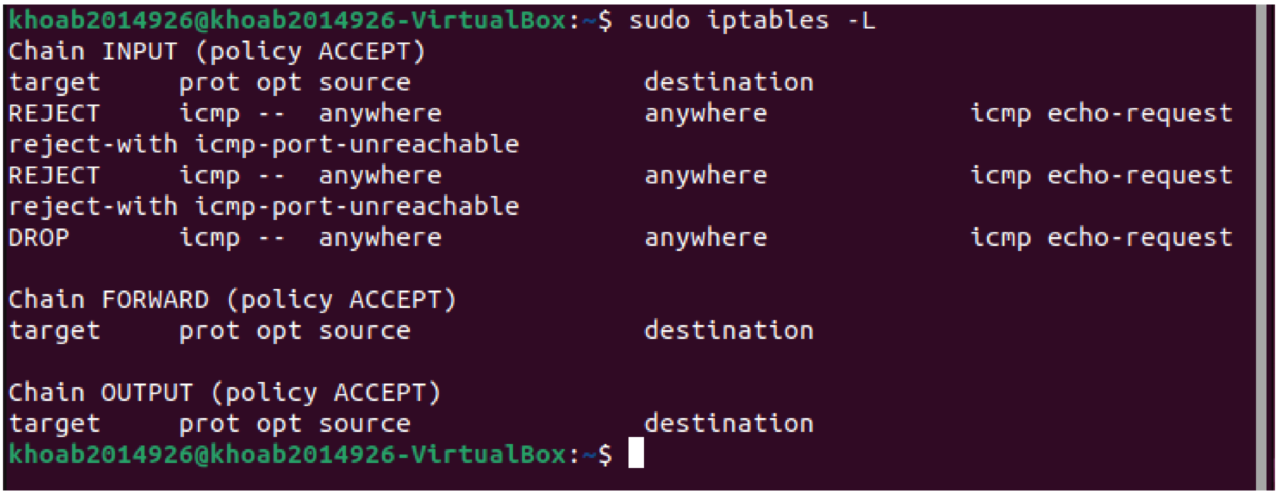


```
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -A INPUT -p icmp --icmp-ty
pe echo-request -j DROP
khoab2014926@khoab2014926-VirtualBox:~$
```

Allow Ping

$ sudo iptables -L

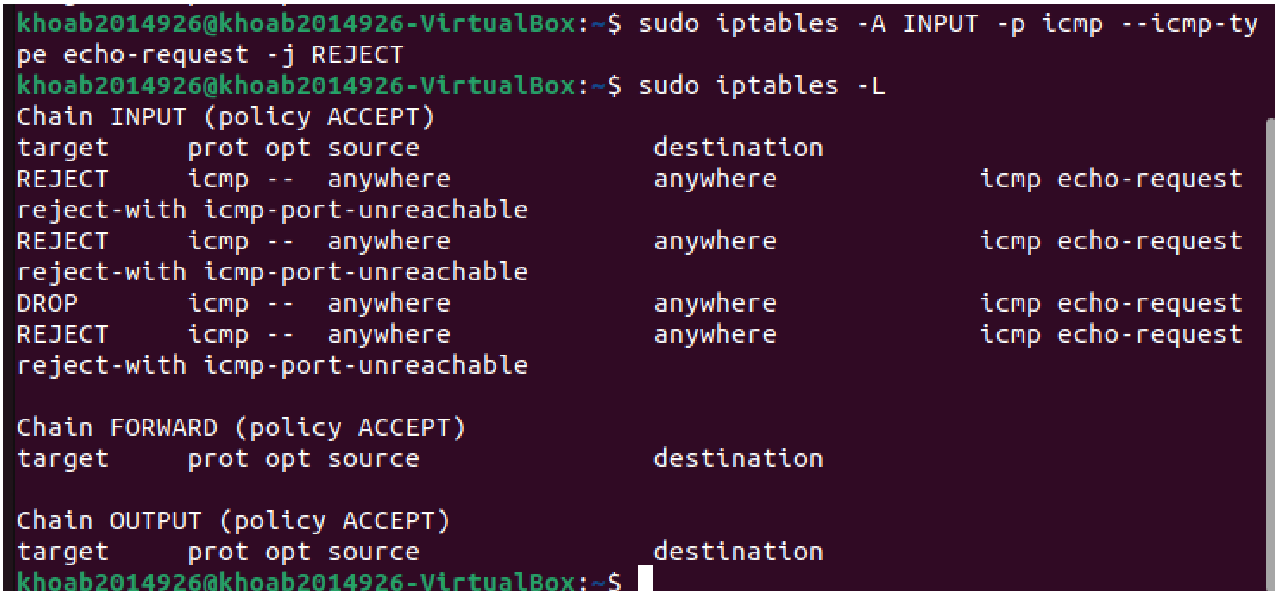
Explain: The command sudo iptables -L lists all of the current iptables rules



```
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
REJECT     icmp --  anywhere             anywhere             icmp echo-request
reject-with icmp-port-unreachable
REJECT     icmp --  anywhere             anywhere             icmp echo-request
reject-with icmp-port-unreachable
DROP       icmp --  anywhere             anywhere             icmp echo-request

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
khoab2014926@khoab2014926-VirtualBox:~$ 
```

⑧ If any of the rules is blocking ping (in our case ICMP is rejected), you can simply remove that rule as follows:

$ sudo iptables -D INPUT -p icmp --icmp-type echo-request -j REJECT

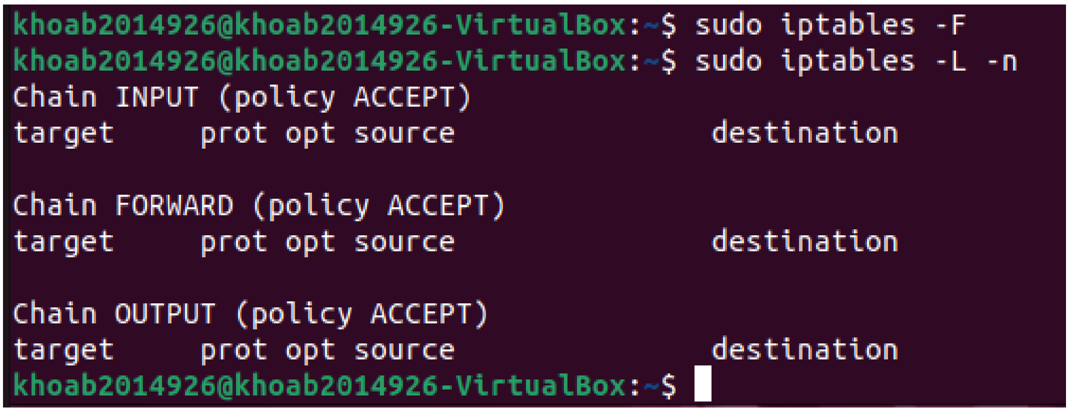


```
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -A INPUT -p icmp --icmp-ty
pe echo-request -j REJECT
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
REJECT     icmp --  anywhere             anywhere             icmp echo-request
reject-with icmp-port-unreachable
REJECT     icmp --  anywhere             anywhere             icmp echo-request
reject-with icmp-port-unreachable
DROP       icmp --  anywhere             anywhere             icmp echo-request
REJECT     icmp --  anywhere             anywhere             icmp echo-request
reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
khoab2014926@khoab2014926-VirtualBox:~$ 
```

⑨ Delete all custom rules added to your iptables Firewall

$ sudo iptables -F

Explain: The command sudo iptables -F flushes all of the current iptables rules



```
khoab2014926@khoab2014926-VirtualBox:~$ sudo iptables -F
khoab2014926@khoab2014926-VirtualBox:~$ sudo 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
khoab2014926@khoab2014926-VirtualBox:~$ 
```