Design of Secure Computer Systems

Lab 07

IPTABLE & TCPIP

The first LAB will illustrate the use of iptables to limit which application services, (ports), will be forwarded through a component serving as a firewall. The second LAB will demonstrate several attacks on the TCP protocol, including the SYN flood attack, the TCP reset attack, and the TCP session hijacking attack.

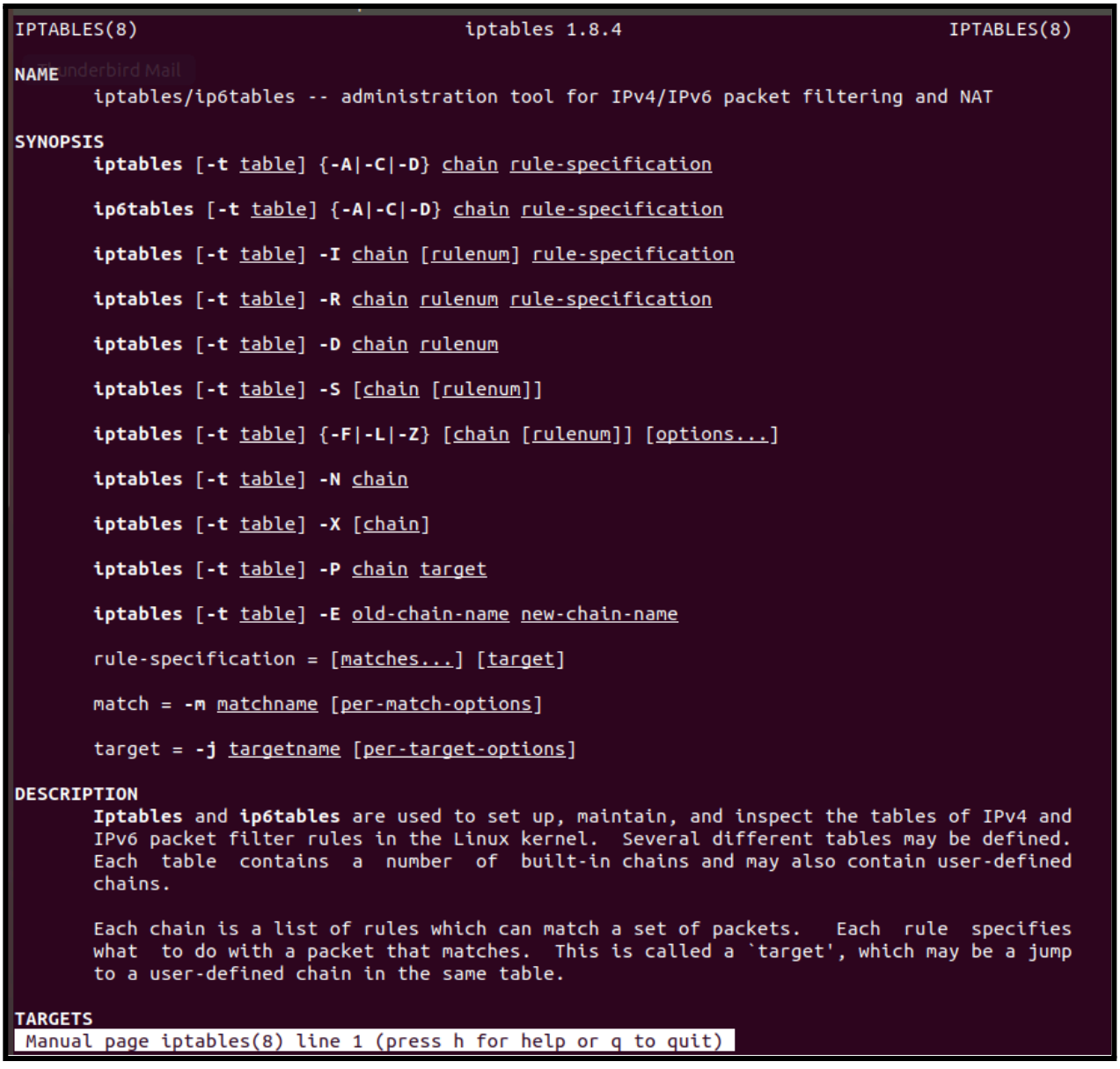
Name: Rakshita Mathur

IP TABLES

The iptables utility is installed on the "firewall" component. Use it to prevent the firewall from forwarding any traffic to the server other than SSH and HTTP sessions.

Following is the demonstration that we have done this by running this command on the client computer: **nmap -n 172.25.0.3** the resulting display should indicate that SSH and HTTP are the only ports that are open.

Looking at the manual by using the man command.



```
ubuntu@client:~$ sudo iptables  -L
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
```

```
ubuntu@client:~$ ls
wizbang
ubuntu@client:~$ sudo nmap 172.25.0.3
Starting Nmap 7.80 ( https://nmap.org ) at 2021-11-05 16:55 UTC
Nmap scan report for server (172.25.0.3)
Host is up (0.000022s latency).
Not shown: 997 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 0.33 seconds
ubuntu@client:~$
```

```
ubuntu@client:~$ sudo iptables -A FORWARD -p tcp --dport 22 -d 172.25.0.3 -j ACCEPT
ubuntu@client:~$ sudo iptables -A FORWARD -p tcp --dport 80 -d 172.25.0.3 -j ACCEPT
ubuntu@client:~$ sudo iptables -A FORWARD -d 172.25.0.3 -j DROP
ubuntu@client:~$ sudo iptables  -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             server              tcp dpt:ssh
ACCEPT     tcp  --  anywhere             server              tcp dpt:http
DROP       all  --  anywhere             server

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

```
ubuntu@client:~$ sudo iptables-save
# Generated by iptables-save v1.8.4 on Fri Nov  5 17:00:13 2021
*nat
:PREROUTING ACCEPT [10:1362]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [1004:44152]
:POSTROUTING ACCEPT [1004:44152]
:DOCKER_OUTPUT - [0:0]
:DOCKER_POSTROUTING - [0:0]
-A OUTPUT -d 127.0.0.11/32 -j DOCKER_OUTPUT
-A POSTROUTING -d 127.0.0.11/32 -j DOCKER_POSTROUTING
-A DOCKER_OUTPUT -d 127.0.0.11/32 -p tcp -m tcp --dport 53 -j DNAT --to-destination 127.0.0.11:44
665
-A DOCKER_OUTPUT -d 127.0.0.11/32 -p udp -m udp --dport 53 -j DNAT --to-destination 127.0.0.11:54
588
-A DOCKER_POSTROUTING -s 127.0.0.11/32 -p tcp -m tcp --sport 44665 -j SNAT --to-source :53
-A DOCKER_POSTROUTING -s 127.0.0.11/32 -p udp -m udp --sport 54588 -j SNAT --to-source :53
COMMIT
# Completed on Fri Nov  5 17:00:13 2021
# Generated by iptables-save v1.8.4 on Fri Nov  5 17:00:13 2021
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A FORWARD -d 172.25.0.3/32 -p tcp -m tcp --dport 22 -j ACCEPT
-A FORWARD -d 172.25.0.3/32 -p tcp -m tcp --dport 80 -j ACCEPT
-A FORWARD -d 172.25.0.3/32 -j DROP
COMMIT
# Completed on Fri Nov  5 17:00:13 2021
ubuntu@client:~$
```

ubuntu@firewall: ~ | ubuntu@firewall: ~

GNU nano 4.8 example_fw.sh

```
#!/bin/bash
#
#  This example IPTABLES firewall will only allow SSH traffic
#  to be forwarded
#
IPTABLES=/sbin/iptables

#start and flush
$IPTABLES -F
$IPTABLES -t nat -F
$IPTABLES -X
#
#  By default, do not allow any forwarding or accept any traffic
#  destined for the firewall.
#
$IPTABLES -P FORWARD DROP
$IPTABLES -P INPUT   DROP
$IPTABLES -P OUTPUT  DROP

# Allow forwarding of traffic associated with any established session
$IPTABLES -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

# Allow SSH traffic on port 22
$IPTABLES -A FORWARD -p tcp --dport 22 -j ACCEPT

# loopback device (internal traffic)
iptables -A INPUT -i lo -p all -j ACCEPT

# log IPTABLES filtering actions
iptables -A FORWARD -j NFLOG -m limit --limit 2/min --nflog-prefix "IPTABLES DROPPED"
```

```
  GNU nano 4.8                                example_fw.sh
#!/bin/bash
#
#  This example IPTABLES firewall will only allow SSH traffic
#  to be forwarded
#
IPTABLES=/sbin/iptables

#start and flush
$IPTABLES -F
$IPTABLES -t nat -F
$IPTABLES -X
#
#  By default, do not allow any forwarding or accept any traffic
#  destined for the firewall.
#
$IPTABLES -P FORWARD DROP
$IPTABLES -P INPUT   DROP
$IPTABLES -P OUTPUT  DROP

# Allow forwarding of traffic associated with any established session
$IPTABLES -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

# Allow SSH traffic on port 22
$IPTABLES -A FORWARD -p tcp --dport 22 -j ACCEPT
$IPTABLES -A FORWARD -p tcp --dport 80 -j ACCEPT
$IPTABLES -A FORWARD -j NFLOG -m limit --limit 2/min --nflog-prefix "IPTABELS DROPPED"
$IPTABLES -A FROWARD -j DROP


# loopback device (internal traffic)
iptables -A INPUT -i lo -p all -j ACCEPT

# log IPTABLES filtering actions
iptables -A FORWARD -j NFLOG -m limit --limit 2/min --nflog-prefix "IPTABLES DROPPED"
```

```
ubuntu@client:~$ wget 172.25.0.3
--2021-11-05 17:35:51--  http://172.25.0.3/
Connecting to 172.25.0.3:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 874 [text/html]
Saving to: 'index.html'

index.html               100%[=============================>]     874  --.-KB/s    in 0s

2021-11-05 17:35:51 (9.51 MB/s) - 'index.html' saved [874/874]

ubuntu@client:~$ ssh 172.25.0.3
kex_exchange_identification: read: Connection reset by peer
ubuntu@client:~$ ssh 172.25.0.3
kex_exchange_identification: read: Connection reset by peer
ubuntu@client:~$ telnet 172.25.0.3
Trying 172.25.0.3...
Connected to 172.25.0.3.
Escape character is '^]'.
Ubuntu 20.04.2 LTS
server login: ubuntu
Password:

Login incorrect
server login:
```

```
ubuntu@firewall:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
NFLOG      all  --  anywhere             anywhere             limit: avg 2/min burst 5 nflog-pref
ix  "IPTABELS DROPPED"
NFLOG      all  --  anywhere             anywhere             limit: avg 2/min burst 5 nflog-pref
ix  "IPTABLES DROPPED"

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ubuntu@firewall:~$
```

```
ubuntu@client:~$ sudo nmap 172.25.0.3
Starting Nmap 7.80 ( https://nmap.org ) at 2021-11-05 17:50 UTC
Nmap scan report for server (172.25.0.3)
Host is up (0.00025s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 4.26 seconds
ubuntu@client:~$
```

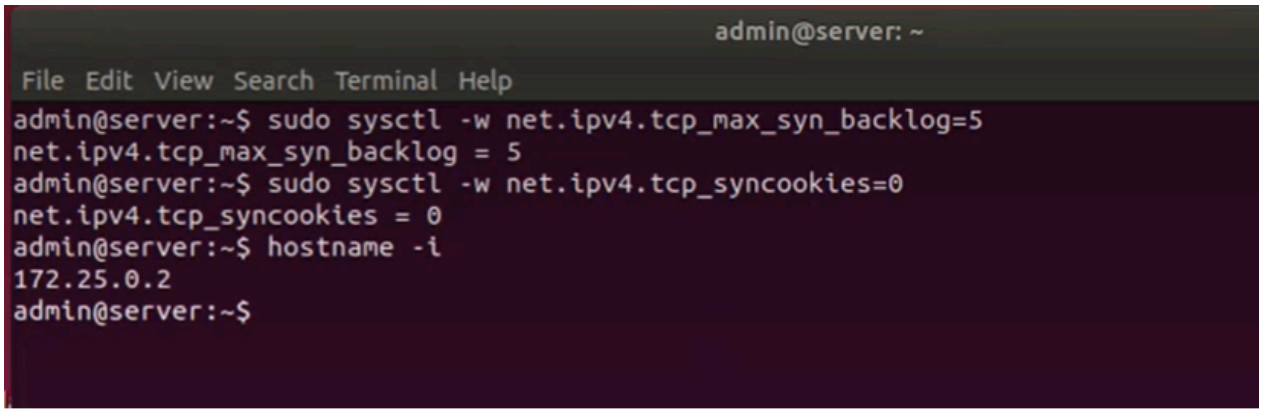
```
sudo: ./wizbnag: command not found
ubuntu@client:~$ cat wizbang
#!/usr/bin/env python
'''
Dumb client interacting with a dumb server
'''
import socket
import sys
import signal
def signal_handler(sig, frame):
    print('Interrupted, exiting')
    sys.exit(0)
try:
    import ssl
except ImportError:
    pass
else:
    signal.signal(signal.SIGINT, signal_handler)
    SERVICE_PORT = 10054
    if len(sys.argv) < 2:
        print('ERROR: missing argument.  Usage:')
        print('    ./wizbang <instruction>')
        exit(1)
    server_host = '172.25.0.3'
    instruction = ' '.join(sys.argv[1:])
    this_host = socket.gethostname()
    conn = socket.socket(socket.AF_INET)
    try:
        conn.connect((server_host, SERVICE_PORT))
    except socket.error as ss:
        print('ERROR: %s' % str(ss))
        exit(1)
    print('Sending instruction %s' % instruction)
    conn.sendall('%s\n' % instruction)
    print('bye')
    conn.close()
```

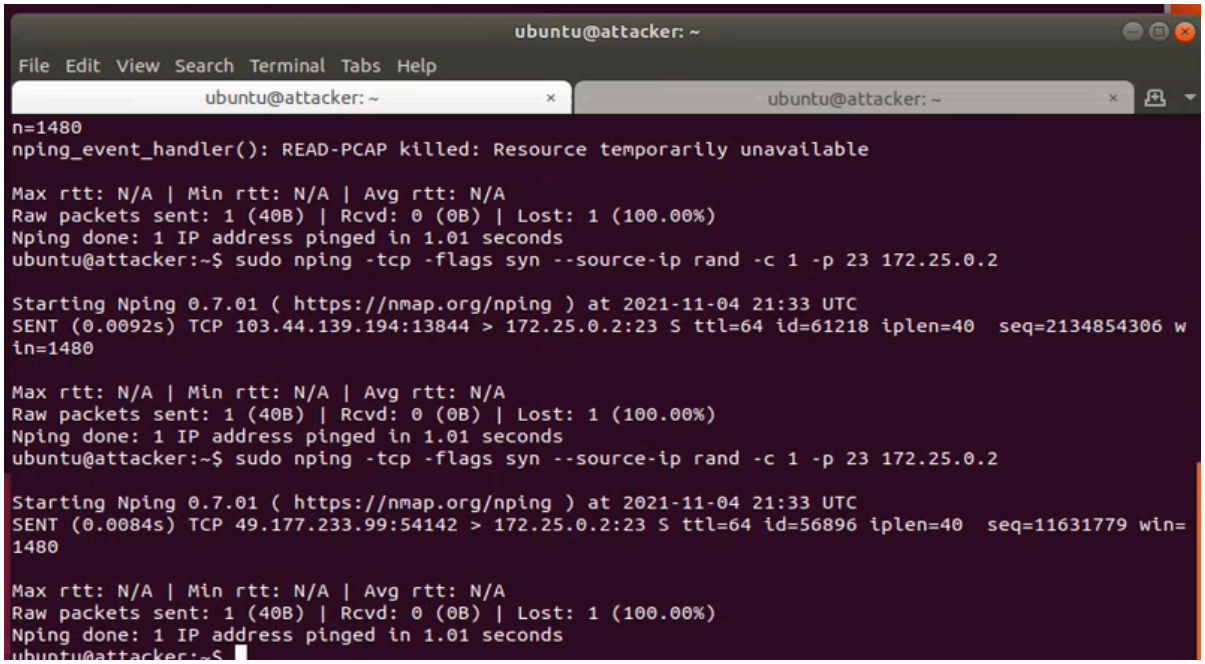
```
 GNU nano 4.8                          example_fw.sh
#!/bin/bash
#
#  This example IPTABLES firewall will only allow SSH traffic
#  to be forwarded
#
IPTABLES=/sbin/iptables

#start and flush
$IPTABLES -F
$IPTABLES -t nat -F
$IPTABLES -X
#
#  By default, do not allow any forwarding or accept any traffic
#  destined for the firewall.
#
$IPTABLES -P FORWARD DROP
$IPTABLES -P INPUT   DROP
$IPTABLES -P OUTPUT  DROP

# Allow forwarding of traffic associated with any established session
$IPTABLES -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

# Allow SSH traffic on port 22
$IPTABLES -A FORWARD -p tcp --dport 22 -j ACCEPT
$IPTABLES -A FORWARD -p tcp --dport 10054 -j ACCEPT
$IPTABLES -A FORWARD -p tcp --dport 80 -j ACCEPT
$IPTABLES -A FORWARD -j NFLOG -m limit --limit 2/min --nflog-prefix "IPTABELS DROPPED"
$IPTABLES -A FROWARD -j DROP


# loopback device (internal traffic)
iptables -A INPUT -i lo -p all -j ACCEPT

# log IPTABLES filtering actions
iptables -A FORWARD -j NFLOG -m limit --limit 2/min --nflog-prefix "IPTABLES DROPPED"
```

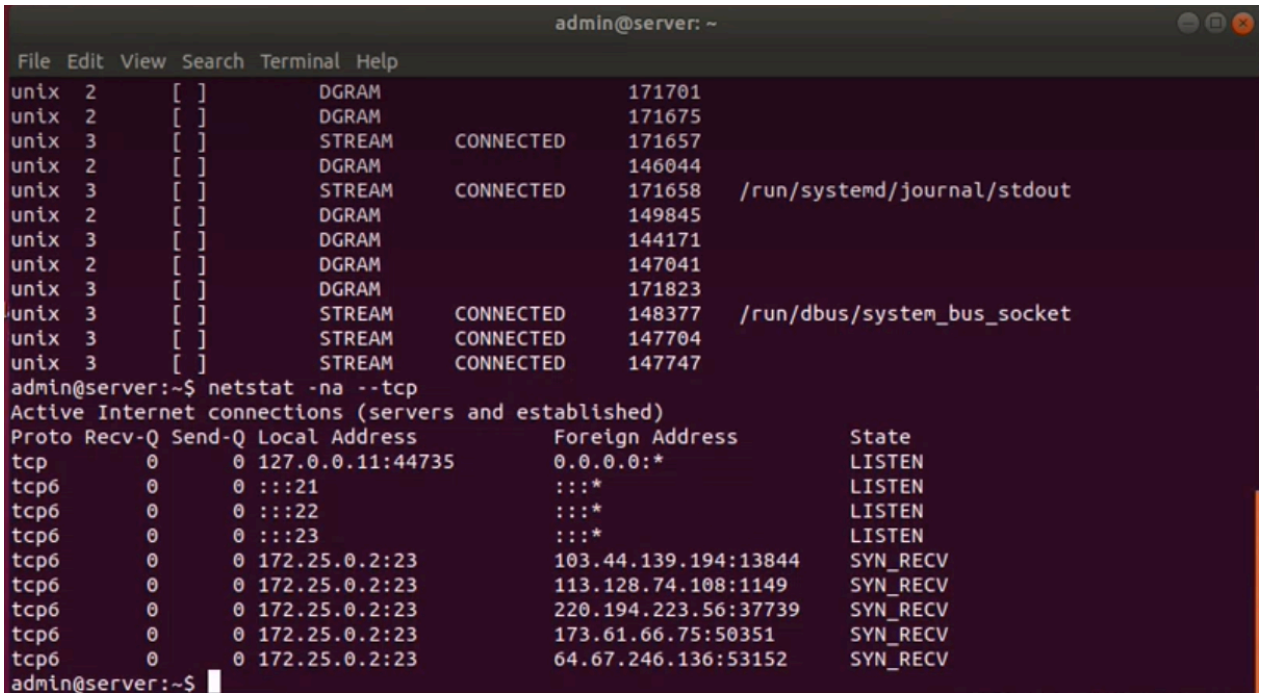
```
ubuntu@client:~$ wizbang run
-bash: wizbang: command not found
ubuntu@client:~$ ./wizbang run
Sending instruction run
bye
ubuntu@client:~$
```

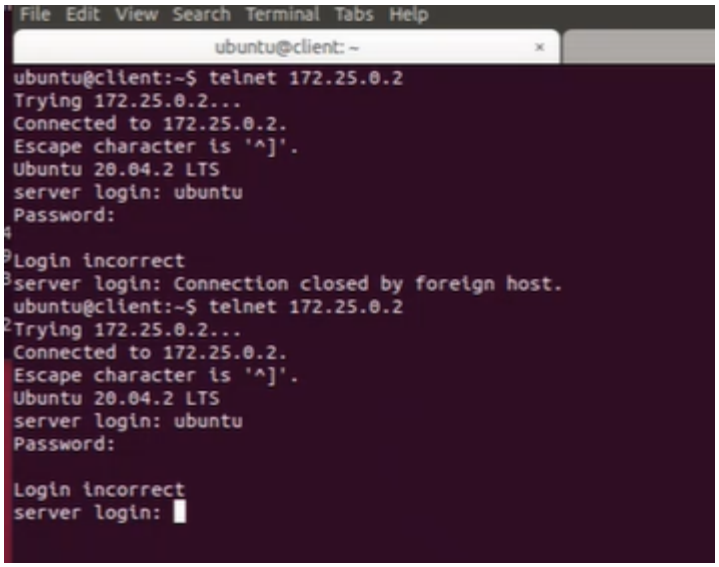
```
student@LabtainersVM:~/labtainer/labtainer-student$ checkwork
Results stored in directory: /home/student/labtainer_xfer/iptables2
Labname iptables2

Student              |  first_ports_ok | second_ports_ok |
==================== | =============== | =============== |
rmath049_at_uottawa. |               Y |               Y |
What is automatically assessed for this lab:

        first_ports_ok: ssh & http were open, and telnet was closed
        second_ports_ok: ssh & http and wizbang were open, telnet closed
```

TCPIP


admin@server: ~
File Edit View Search Terminal Help
admin@server:~$ sudo sysctl -w net.ipv4.tcp_max_syn_backlog=5
net.ipv4.tcp_max_syn_backlog = 5
admin@server:~$ sudo sysctl -w net.ipv4.tcp_syncookies=0
net.ipv4.tcp_syncookies = 0
admin@server:~$ hostname -i
172.25.0.2
admin@server:~$


ubuntu@attacker: ~
File Edit View Search Terminal Tabs Help
ubuntu@attacker: ~
ubuntu@attacker: ~
n=1480
nping_event_handler(): READ-PCAP killed: Resource temporarily unavailable
Max rtt: N/A | Min rtt: N/A | Avg rtt: N/A
Raw packets sent: 1 (40B) | Rcvd: 0 (0B) | Lost: 1 (100.00%)
Nping done: 1 IP address pinged in 1.01 seconds
ubuntu@attacker:~$ sudo nping -tcp -flags syn --source-ip rand -c 1 -p 23 172.25.0.2
Starting Nping 0.7.01 (https://nmap.org/nping) at 2021-11-04 21:33 UTC
SENT (0.0092s) TCP 103.44.139.194:13844 > 172.25.0.2:23 S ttl=64 id=61218 iplen=40 seq=2134854306 w
in=1480
Max rtt: N/A | Min rtt: N/A | Avg rtt: N/A
Raw packets sent: 1 (40B) | Rcvd: 0 (0B) | Lost: 1 (100.00%)
Nping done: 1 IP address pinged in 1.01 seconds
ubuntu@attacker:~$ sudo nping -tcp -flags syn --source-ip rand -c 1 -p 23 172.25.0.2
Starting Nping 0.7.01 (https://nmap.org/nping) at 2021-11-04 21:33 UTC
SENT (0.0084s) TCP 49.177.233.99:54142 > 172.25.0.2:23 S ttl=64 id=56896 iplen=40 seq=11631779 win=
1480
Max rtt: N/A | Min rtt: N/A | Avg rtt: N/A
Raw packets sent: 1 (40B) | Rcvd: 0 (0B) | Lost: 1 (100.00%)
Nping done: 1 IP address pinged in 1.01 seconds





Was unable to connect to telnet, therefore, could not do the rest of the lab.