SUBJECT:**COMPUTER NETWORKS(CA-2 PACKAGE)**

TOPIC : TCP PORT SCAN OVER ICMP(PING) RESPONDED HOSTS

TEAM MEMBERS:
Dhikshitha.A(19PD09)
Swathi Prathaa.P(19PD38)

ABSTRACT : The basic idea behind our package is to find the hosts that are online and run a TCP port scan over these hosts.
Our program does two operations namely:

1. Pinging the subnet of hosts to find the ones that are online,storing its count and sending these hosts one by one to the tcp_scan function.
2. The tcp_scan function has a set of TCP ports that are predefined.The responded hosts are tcp scanned to find the no.of tcp ports that are open for that particular host and a count is returned.

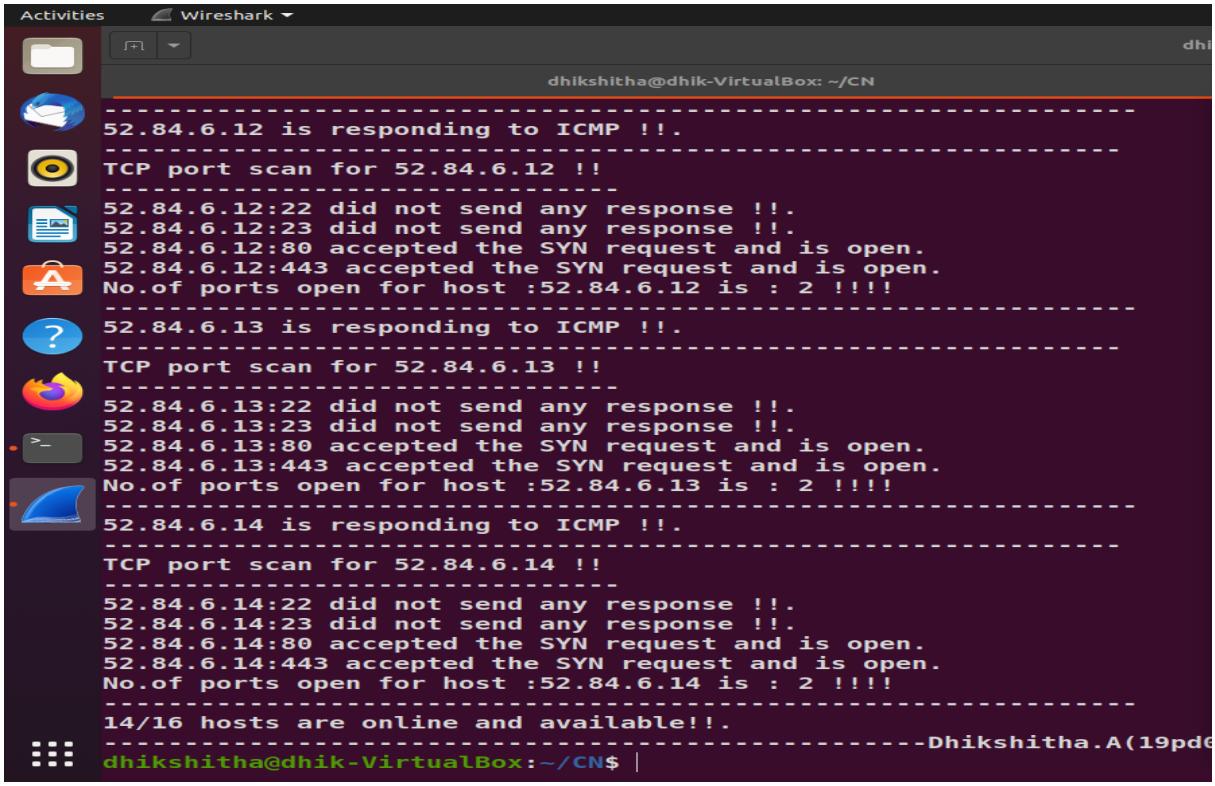
So the final output will have the following

- subnet of hosts scanned.
- The count of hosts that are online.
- If the host has responded then the no.of tcp ports open for that host.

OUTPUT:

Case 1: Wireshark capture of the host's that responded to the ping request

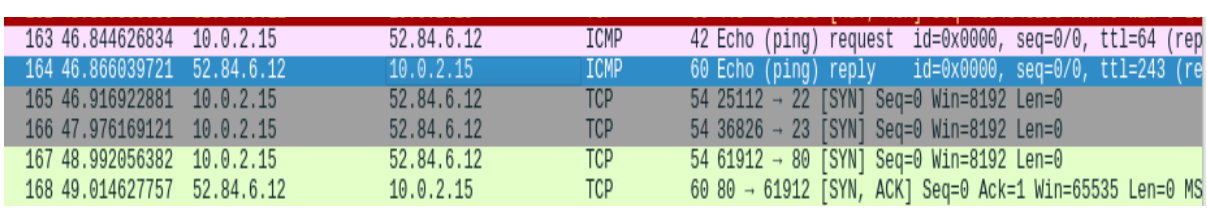
In the below image,host:52.84.6.12,host:52.84.6.13 and host:52.84.6.14 has responded to the ping request and TCP port scan has been done for these hosts.As you can see from the screenshot,2 ports are open for these hosts.



In the above case all the 14/16 hosts responded(num_addresses-The total number of addresses in the network which is 16 addresses in our case out of which 14 are hosts and the remaining 2 are network and broadcast addresses).

Wireshark capture for the ip is shown below. As you can see for the ping(ICMP)-request its corresponding ping-reply has been captured.TCP port scanning has been done next. SYN requests(request for connection) have been sent for all ports 22,23,80 and 443 but only ports 80(shown below) and 443 responded with SYN-ACK.

host:52.84.6.12



163	46.844626834	10.0.2.15	52.84.6.12	ICMP	42 Echo (ping) request id=0x0000, seq=0/0, ttl=64 (rep
164	46.866039721	52.84.6.12	10.0.2.15	ICMP	60 Echo (ping) reply id=0x0000, seq=0/0, ttl=243 (re
165	46.916922881	10.0.2.15	52.84.6.12	TCP	54 25112 → 22 [SYN] Seq=0 Win=8192 Len=0
166	47.976169121	10.0.2.15	52.84.6.12	TCP	54 36826 → 23 [SYN] Seq=0 Win=8192 Len=0
167	48.992056382	10.0.2.15	52.84.6.12	TCP	54 61912 → 80 [SYN] Seq=0 Win=8192 Len=0
168	49.014627757	52.84.6.12	10.0.2.15	TCP	60 80 → 61912 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MS

host:52.84.6.13:

177	51.217739217	10.0.2.15	52.84.6.13	ICMP	42	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (rep
178	51.239686284	52.84.6.13	10.0.2.15	ICMP	60	Echo (ping) reply id=0x0000, seq=0/0, ttl=243 (re
179	51.301050795	10.0.2.15	52.84.6.13	TCP	54	7135 → 22 [SYN] Seq=0 Win=8192 Len=0
180	52.329196035	10.0.2.15	52.84.6.13	TCP	54	10627 → 23 [SYN] Seq=0 Win=8192 Len=0
181	53.377703292	10.0.2.15	52.84.6.13	TCP	54	29876 → 80 [SYN] Seq=0 Win=8192 Len=0
182	53.401323273	52.84.6.13	10.0.2.15	TCP	60	80 → 29876 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MS

186	54.458086204	10.0.2.15	52.84.6.13	TCP	54	58418 → 443 [SYN] Seq=0 Win=8192 Len=0
187	54.480969115	52.84.6.13	10.0.2.15	TCP	60	443 → 58418 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 M

Case 2:1 out of 14 hosts did not respond to the ping request

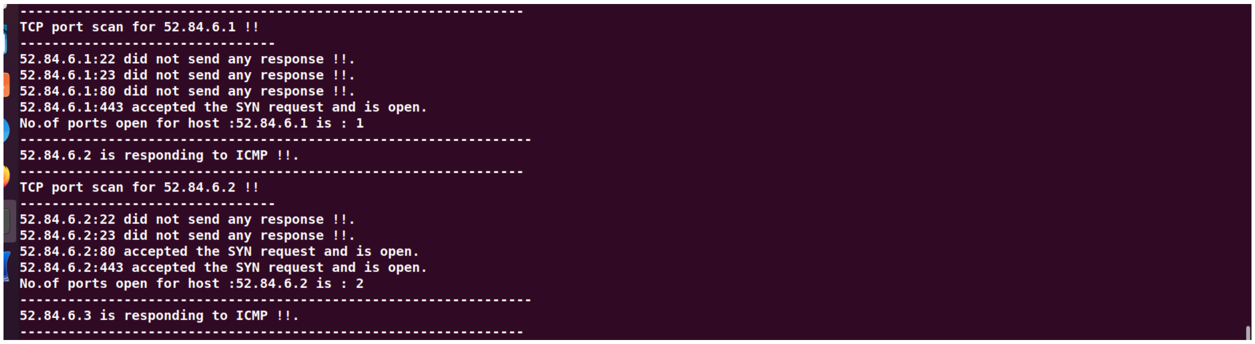
Host:52.84.6.7 did not respond to the ping request(52.84.6.7 is down or not responding).

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
52.84.6.7 is down or not responding!!!!.
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
52.84.6.8 is responding to ICMP !!.
---------------------------------------------------------------
TCP port scan for 52.84.6.8 !!
-------------------------------
52.84.6.8:22 did not send any response !!.
52.84.6.8:23 did not send any response !!.
52.84.6.8:80 accepted the SYN request and is open.
52.84.6.8:443 accepted the SYN request and is open.
No.of ports open for host :52.84.6.8 is : 2 !!!!
---------------------------------------------------------------
52.84.6.9 is responding to ICMP !!.
---------------------------------------------------------------
TCP port scan for 52.84.6.9 !!
-------------------------------
52.84.6.9:22 did not send any response !!.
52.84.6.9:23 did not send any response !!.
52.84.6.9:80 accepted the SYN request and is open.
```

```
TCP port scan for 52.84.6.14 !!
-------------------------------
52.84.6.14:22 did not send any response !!.
52.84.6.14:23 did not send any response !!.
52.84.6.14:80 accepted the SYN request and is open.
52.84.6.14:443 accepted the SYN request and is open.
No.of ports open for host :52.84.6.14 is : 2 !!!!
---------------------------------------------------------------
13/16 hosts are online and available!!.
---------------------------------------------------------------Dhikshitha.A(19pd09) and Swathi Prathaa.P(19pd38)---------------------------------------
```

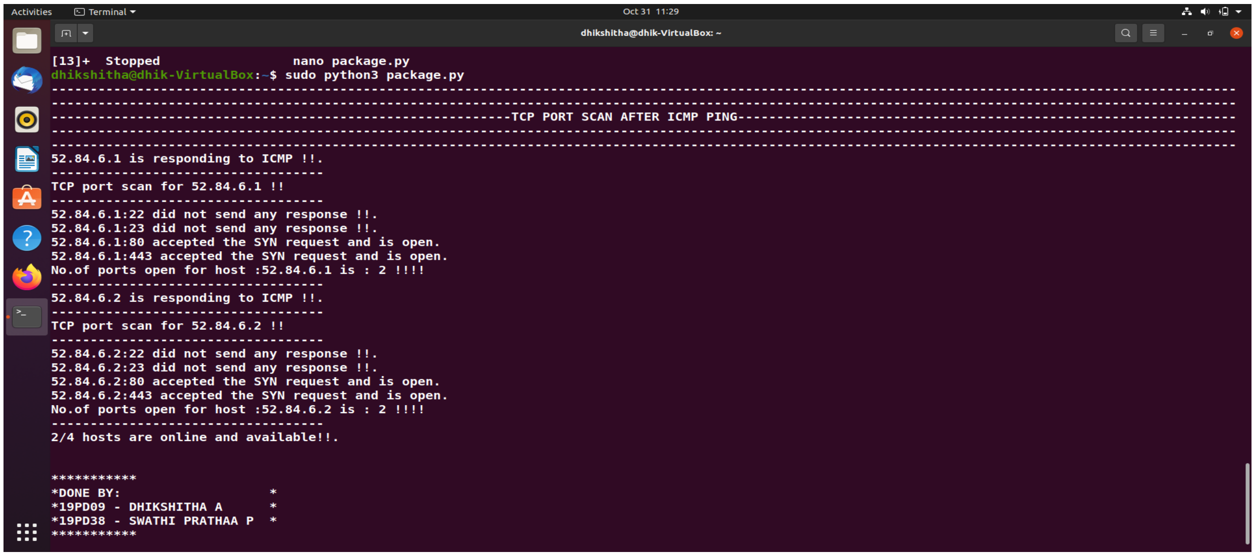
Case 3:For a particular host only one TCP port was open:

For the host:52.84.6.1 only one port was open. So the count for the no.of ports open for that host is 1.



```
----------------------------------------------------------------
TCP port scan for 52.84.6.1 !!
--------------------------------
52.84.6.1:22 did not send any response !!.
52.84.6.1:23 did not send any response !!.
52.84.6.1:80 did not send any response !!.
52.84.6.1:443 accepted the SYN request and is open.
No.of ports open for host :52.84.6.1 is : 1
-----------------------------------------------------------------
52.84.6.2 is responding to ICMP !!.
----------------------------------------------------------------
TCP port scan for 52.84.6.2 !!
--------------------------------
52.84.6.2:22 did not send any response !!.
52.84.6.2:23 did not send any response !!.
52.84.6.2:80 accepted the SYN request and is open.
52.84.6.2:443 accepted the SYN request and is open.
No.of ports open for host :52.84.6.2 is : 2
-----------------------------------------------------------------
52.84.6.3 is responding to ICMP !!.
----------------------------------------------------------------
```

Case 4:Changing the prefix value to net="52.84.6.0/30":



```
[13]+  Stopped                 nano package.py
dhikshitha@dhik-VirtualBox:~$ sudo python3 package.py
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------TCP PORT SCAN AFTER ICMP PING----------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
52.84.6.1 is responding to ICMP !!.
-----------------------------------
TCP port scan for 52.84.6.1 !!
-----------------------------------
52.84.6.1:22 did not send any response !!.
52.84.6.1:23 did not send any response !!.
52.84.6.1:80 accepted the SYN request and is open.
52.84.6.1:443 accepted the SYN request and is open.
No.of ports open for host :52.84.6.1 is : 2 !!!!
-----------------------------------
52.84.6.2 is responding to ICMP !!.
-----------------------------------
TCP port scan for 52.84.6.2 !!
-----------------------------------
52.84.6.2:22 did not send any response !!.
52.84.6.2:23 did not send any response !!.
52.84.6.2:80 accepted the SYN request and is open.
52.84.6.2:443 accepted the SYN request and is open.
No.of ports open for host :52.84.6.2 is : 2 !!!!
-----------------------------------
2/4 hosts are online and available!!.


***********
*DONE BY:                   *
*19PD09 - DHIKSHITHA A      *
*19PD38 - SWATHI PRATHAA P  *
***********
```

In the above case all the 2/4 hosts responded (num_addresses- The total number of addresses in the network which is 4 addresses in the above case and the remaining 2 are network and broadcast addresses).

Code:

```
###Performing a tcp port scan over hosts that are responding
to ping request
from ipaddress import IPv4Network
from typing import List
from scapy.all import ICMP, IP, sr1, TCP
import random
from scapy.all import *
```

```
print("-------------------------------------------------------
-------------------------------------------------------------
--------->
print("-------------------------------------------------------
-------------------------------------------------------------
--------->
print("-------------------------------------------------------
-----TCP PORT SCAN AFTER ICMP
PING------------------------------------->
print("-------------------------------------------------------
-------------------------------------------------------------
--------->
print("-------------------------------------------------------
-------------------------------------------------------------
--------->


##performing tcp-port scan to find open ports for responding
hosts in the network
def tcp_scan(host: str, ports: List[int]):
    count1 = 0

    for dstp in ports:
        srcp = random.randint(1025, 65534)
        response = sr1(
            IP(dst=host)/TCP(sport=srcp, dport=dstp,
flags="S"), timeout=1,
            verbose=0,
        )
        if response is None:
            print(f"{host}:{dstp} did not send any response
!!.")

        elif(response.haslayer(TCP)):
            if(response.getlayer(TCP).flags == 0x12):
                send_rst = sr1(
  IP(dst=host)/TCP(sport=srcp, dport=dstp, flags='R'),
                    timeout=1,
```

```
                    verbose=0,
                )
                count1 = count1+1
                print(f"{host}:{dstp} accepted the SYN
request and is open.")

            elif (response.getlayer(TCP).flags == 0x14):
                print(f"{host}:{dstp} is closed!!!.")

    print(f"No.of ports open for host :{host} is :",
count1,"!!!!")



##defining a subnet of hosts
net = "52.84.6.0/30"

tcp_port_list = [22, 23, 80, 443]


#try:
#       addr=IPv4Network(net)
#except:
#       addr.AddressValueError(ValueError)
#       addr.NetmaskValueError(ValueError)


addr=IPv4Network(net)
count = 0
##performing icmp ping to check for ip's that respond to the
request

for host in addr:
    if (host in (addr.network_address,
addr.broadcast_address)):
        # Skip network and broadcast addresses
        continue
```

```
    response = sr1(IP(dst=str(host))/ICMP(), timeout=2,
verbose=0)

    if response is None:
        print("xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx")
        print(f"{host} is down or not responding!!!!.")
        print("xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx")
    elif (
        # type3-destination unreachable
        int(response.getlayer(ICMP).type) == 3 and
        int(response.getlayer(ICMP).code) in [1, 2, 3, 9, 10,
13]
    ):
        print(f"{host} is blocking ICMP.")
    else:

        print(f"{host} is responding to ICMP !!.")
        print("-----------------------------------")
        print(f"TCP port scan for {host} !!")
        print("-----------------------------------")
        tcp_scan(str(host), tcp_port_list)
        print("-----------------------------------")
        count += 1

print(f"{count}/{addr.num_addresses} hosts are online and
available!!.")
print("\n")
print("***********")
print("*DONE BY:                  *")
print("*19PD09 - DHIKSHITHA A     *")
print("*19PD38 - SWATHI PRATHAA P  *")
print("***********")
print("\n")
```

CONTRIBUTION :

Swathi Prathaa - Pinging hosts
Dhikshitha - Tcp port scan over pinged hosts
(Both of us contributed equally for the research and implementation of this project.)

To run:

```
autopep8 -i filename.py
sudo python3 filename.py
```