Assignment 2.1
ITE 423 – Information Security

2016043654 경제금융학부 이창모

<Task-1>

* Environment: python code on Windows(host os). test target: kali-linux(guest os)

Before I start, I tested with my virtual machine. IP address of kali-linux is 192.168.73.128

```
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.73.128  netmask 255.255.255.0  broadcast 192.168.73.255
        inet6 fe80::20c:29ff:fefe:fdb8  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:fe:fd:b8  txqueuelen 1000  (Ethernet)
        RX packets 50  bytes 3756 (3.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 25  bytes 2470 (2.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

I opened tcp 23 port(telnet) on kali-linux.

```
┌──(root💀kali)-[/home/tlqkf]
└─# netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Pro
gram name
tcp6       0      0 :::23                   :::*                    LISTEN      1393/xi
netd
```

The followings are my test source code and the result.

```
target = "192.168.73.128"

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.settimeout(5)
result = s.connect_ex((target, 23))
s.close()

if result:
    print("not open")
else:
    print("open")
```

```
= RESTART: C:/Users/이창모/OneDrive - 한양대학교/바탕 화
ang.py =
==================start================
open
```

It works well. However, when I tried with UDP scan, I works poor.

Now, get down to work.

code:

/* scan_hanyang.py */

```
# 166.104.177.24: www.hanyang.ac.kr
# HYU uses class B. /16
import socket
import time

def TCPscan():
    print("="*10 + "TCP 80/8080 scan start" + "="*10)

    global target # target IP
    global TCPnum # to count total number of web servers

    for ip_target in range(1, 255): # except network address: 0 and broadcast address: 255
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        socket.setdefaulttimeout(1)

        target += str(ip_target)

        result80 = s.connect_ex((target, 80))
        result8080 = s.connect_ex((target, 8080))

        # find domain_name if exists
        target_domain_name=""
        try:
            target_domain_name = socket.gethostbyaddr(target)[0]
        except Exception as e:
            pass

        if not result80:
            TCPnum += 1
            print(target + " port 80 is open.\t" + target_domain_name)

        if not result8080:
            TCPnum += 1
            print(target + " port 8080 is open.\t" + target_domain_name)

        target = "166.104.177."
        time.sleep(1) # not to be blocked from FW


def UDPscan(): # unreliable result
    print("="*10 + "UDP 80/8080 scan start" + "="*10)
    global target # target IP
    global UDPnum # to count total number of web servers

    for ip_target in range(1, 255): # except network address: 0 and broadcast address: 255
        s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        socket.setdefaulttimeout(1)
```

```
        target += str(ip_target)

        addr80 = (target, 80)
        addr8080 = (target, 8080)
        li = [addr80, addr8080]


        # find domain_name if exists
        target_domain_name=""
        try:
            target_domain_name = socket.gethostbyaddr(target)[0]
        except Exception as e:
            pass


        for i in li:
            try:
                s.sendto(b'hello', i)
                s.recvfrom(1024)
                UDPnum += 1
                print(target + " port " + str(i[1]) + " is open.\t" + target_domain_name)
               # received reply
            except Exception as e:
                if str(e) == "timed out":
                    UDPnum += 1
                    print(target + " port " + str(i[1]) + " is open.\t"
                          + target_domain_name)
                    # no reply
                # else: ICMP unreachable: closed

        target = "166.104.177."
        time.sleep(1) # not to be blocked from FW


#main
if __name__ == "__main__":
    start = time.time()
    target = "166.104.177." # + a
    TCPnum = 0
    UDPnum = 0

    TCPscan()
    print("\n")
    UDPscan()

    print("\n\nTotal number of web servers: TCP-" + str(TCPnum) + " UDP-" + str(UDPnum))
    print("Scan duration: " + str(round(time.time() - start)) + " sec")
```

I also tried UDP scan using the similar code with TCP scan, which uses socket.connect_ex, but the result is same. It returned unreliable result.

result:

```
==========TCP 80/8080 scan start==========
166.104.177.24 port 80 is open.         www.hanyang.ac.kr
166.104.177.30 port 80 is open.
166.104.177.50 port 80 is open.
166.104.177.71 port 80 is open.
166.104.177.72 port 80 is open.
166.104.177.108 port 80 is open.        hanyang.ac.kr
166.104.177.133 port 80 is open.
166.104.177.140 port 80 is open.
166.104.177.142 port 80 is open.
166.104.177.151 port 80 is open.
166.104.177.152 port 80 is open.
166.104.177.155 port 80 is open.
166.104.177.170 port 80 is open.        portal.hanyang.ac.kr
166.104.177.180 port 80 is open.
166.104.177.190 port 80 is open.
166.104.177.191 port 80 is open.
166.104.177.200 port 80 is open.        nf.hanyang.ac.kr
166.104.177.201 port 80 is open.


==========UDP 80/8080 scan start==========
166.104.177.1 port 80 is open.
166.104.177.1 port 8080 is open.
```

(...)skip

```
166.104.177.23 port 8080 is open.
166.104.177.24 port 80 is open.         www.hanyang.ac.kr
166.104.177.24 port 8080 is open.       www.hanyang.ac.kr
166.104.177.25 port 80 is open.
166.104.177.25 port 8080 is open.
166.104.177.26 port 80 is open.
166.104.177.26 port 8080 is open.
166.104.177.27 port 80 is open.
166.104.177.27 port 8080 is open.
166.104.177.28 port 80 is open.
166.104.177.28 port 8080 is open.
166.104.177.29 port 80 is open.
166.104.177.29 port 8080 is open.
166.104.177.30 port 80 is open.
166.104.177.30 port 8080 is open.
166.104.177.31 port 80 is open.
```

(...)skip

166.104.177.102 port 8080 is open.
166.104.177.103 port 80 is open. antispam1.hanyang.ac.kr
166.104.177.103 port 8080 is open. antispam1.hanyang.ac.kr
166.104.177.104 port 80 is open. antispam2.hanyang.ac.kr
166.104.177.104 port 8080 is open. antispam2.hanyang.ac.kr
166.104.177.105 port 80 is open. mail.hanyang.ac.kr
166.104.177.105 port 8080 is open. mail.hanyang.ac.kr
166.104.177.106 port 80 is open. mail.hanyang.ac.kr
166.104.177.106 port 8080 is open. mail.hanyang.ac.kr
166.104.177.107 port 80 is open.
166.104.177.107 port 8080 is open.
166.104.177.108 port 80 is open. hanyang.ac.kr
166.104.177.108 port 8080 is open. hanyang.ac.kr
166.104.177.109 port 80 is open. antispam.hanyang.ac.kr
166.104.177.109 port 8080 is open. antispam.hanyang.ac.kr
166.104.177.110 port 80 is open.
166.104.177.110 port 8080 is open.
166.104.177.111 port 80 is open.

(...)skip

166.104.177.168 port 80 is open.
166.104.177.168 port 8080 is open.
166.104.177.169 port 80 is open.
166.104.177.169 port 8080 is open.
166.104.177.170 port 80 is open. portal.hanyang.ac.kr
166.104.177.170 port 8080 is open. portal.hanyang.ac.kr
166.104.177.171 port 80 is open.
166.104.177.171 port 8080 is open.
166.104.177.172 port 80 is open.
166.104.177.172 port 8080 is open.
166.104.177.173 port 80 is open.
166.104.177.173 port 8080 is open.
166.104.177.174 port 80 is open.

(...)skip

166.104.177.60 port 80 is open.
166.104.177.60 port 8080 is open.
166.104.177.61 port 80 is open. rmail.hanyang.ac.kr
166.104.177.61 port 8080 is open. rmail.hanyang.ac.kr
166.104.177.62 port 80 is open. nmail.hanyang.ac.kr
166.104.177.62 port 8080 is open. nmail.hanyang.ac.kr
166.104.177.63 port 80 is open.
166.104.177.63 port 8080 is open.
166.104.177.64 port 80 is open.
166.104.177.64 port 8080 is open.
166.104.177.65 port 80 is open.
166.104.177.65 port 8080 is open.
166.104.177.66 port 80 is open.
166.104.177.66 port 8080 is open.
166.104.177.67 port 80 is open.

(...)skip

166.104.177.250 port 8080 is open.
166.104.177.251 port 80 is open.
166.104.177.251 port 8080 is open.
166.104.177.252 port 80 is open.
166.104.177.252 port 8080 is open.
166.104.177.253 port 80 is open.
166.104.177.253 port 8080 is open.
166.104.177.254 port 80 is open.
166.104.177.254 port 8080 is open.

Total number of web servers: TCP-18 UDP-508
Scan duration: 3604 sec

<Task-2>

*Environment: os(kali-linux)

The result was too long to show in a page. I just "greped" open port with IP. I added -T2 option not to be blocked from FW.

TCP scan:

```
┌──(root💀kali)-[/home/tlqkf/test]
└─# nmap -sS -p 80,8080 166.104.177.1-254 -T1 >> TCPScanResult.txt
```

result:

```
┌──(root💀kali)-[/home/tlqkf/test]
└─# cat TCPScanResult.txt | grep -P (open|166.104)
```

```
Nmap scan report for www.hanyang.ac.kr (166.104.177.24)
80/tcp   open   http

Nmap scan report for 166.104.177.30
80/tcp   open   http

Nmap scan report for 166.104.177.50
80/tcp   open   http

Nmap scan report for 166.104.177.71
80/tcp   open   http
Nmap scan report for 166.104.177.72
80/tcp   open   http

Nmap scan report for hanyang.ac.kr (166.104.177.108)
80/tcp   open   http

Nmap scan report for 166.104.177.133
80/tcp   open   http

Nmap scan report for 166.104.177.140
80/tcp   open   http

Nmap scan report for 166.104.177.142
80/tcp   open   http
Nmap scan report for 166.104.177.151
80/tcp   open   http

Nmap scan report for 166.104.177.152
80/tcp   open   http
Nmap scan report for 166.104.177.155
80/tcp   open   http
Nmap scan report for portal.hanyang.ac.kr (166.104.177.170)
80/tcp   open   http
Nmap scan report for 166.104.177.180
80/tcp   open   http

Nmap scan report for 166.104.177.190
80/tcp   open   http
Nmap scan report for 166.104.177.191
80/tcp   open   http
Nmap scan report for nf.hanyang.ac.kr (166.104.177.200)
80/tcp   open   http
Nmap scan report for 166.104.177.201
80/tcp   open  http
8080/tcp open  http-proxy
```

The only difference with the result of the code(Task-1) is that nmap shows 166.104.177.201 opens 8080.

UDP scan:

```
┌──(root💀kali)-[/home/tlqkf/test]
└─# nmap -sU -p 80,8080 166.104.177.1-254 -T2 >> UDPScanResult.txt
```

result:

```
┌──(root💀kali)-[/home/tlqkf/test]
└─# more UDPScanResult.txt | grep -P "166.104|open"
Nmap scan report for 166.104.177.1
80/udp   open|filtered http
8080/udp open|filtered http-alt
Nmap scan report for 166.104.177.2
80/udp   open|filtered http
8080/udp open|filtered http-alt
Nmap scan report for 166.104.177.3
80/udp   open|filtered http
8080/udp open|filtered http-alt
Nmap scan report for 166.104.177.4
80/udp   open|filtered http
8080/udp open|filtered http-alt
```

From the start to the end, it showed "open|filtered". This is because UDPScan is unreliable. This result Task-2-UDP scan is in line with that of Task-1-UDP scan.

Difficulties:

1. I was blocked by Hanyang University FW, due to "-T2" option which is too fast to scan to avoid FW. I tried several times with "-T1" option after a while. Then It was successful.

```
... skipping 1 line
PORT     STATE    SERVICE
80/tcp   filtered http
8080/tcp closed   http-proxy

Nmap done: 254 IP addresses (254 hosts up) scanned in 1344.62 seconds

┌──(root💀kali)-[/home/tlqkf/test]
└─# more TCPScanResult.txt | grep open
```

Nothing was "greped", as the result was composed of only "filtered" and "closed" with "-T2".

2. As UDP scan is unreliable, the result of UDP scan was strange. I reviewed all of the codes more than 3 times and searched in Google. Finally, I found the answer that can explain my situation(UDP scanning is unreliable ~~). It took a lot of time although my code was not wrong.