

INTERNET PROTOCOL LAB ASSIGNMENT -2

Name: Siriparapu Sparshika

Roll No: CYS22006

Date: 22-10-2022

TASKS:

1. Understand about PING and document it, then answer the following question ping commands uses ICMP protocol to check if the requested resource is responding or not. These are otherwise called as error-checking protocols as they are used by network devices to identify network connectivity issues.

a. Use ping on google.com and document your results on the output you received. [Find the IP address, Time to live value and round-trip time value from the results you got].

```
C:\Windows\system32>ping google.com

Pinging google.com [2404:6800:4007:814::200e] with 32 bytes of data:
Reply from 2404:6800:4007:814::200e: time=102ms
Reply from 2404:6800:4007:814::200e: time=108ms
Reply from 2404:6800:4007:814::200e: time=546ms
Reply from 2404:6800:4007:814::200e: time=41ms

Ping statistics for 2404:6800:4007:814::200e:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 41ms, Maximum = 546ms, Average = 199ms

C:\Windows\system32>
```

IP address: 2404:6800:4007:814::200e

Time to Live (TTL): None (128 default)

Round-trip time: Minimum=41ms, Maximum=546ms, Average=199ms

b. By default, ping will send 4 packets to check the details, here you have to send 8 packets to check the output over google.com. Explain what the purpose of doing this is.

ping command also has ping -n (count) which states number of times the request has to be sent. The purpose of doing so is to ensure that the path can handle mentioned number of packets and to test the stability of the line.

```
C:\Windows\system32>ping -n 8 google.com

Pinging google.com [2404:6800:4009:811::200e] with 32 bytes of data:
Reply from 2404:6800:4009:811::200e: time=104ms
Reply from 2404:6800:4009:811::200e: time=122ms
Reply from 2404:6800:4009:811::200e: time=89ms
Reply from 2404:6800:4009:811::200e: time=114ms
Reply from 2404:6800:4009:811::200e: time=120ms
Reply from 2404:6800:4009:811::200e: time=132ms
Reply from 2404:6800:4009:811::200e: time=117ms
Reply from 2404:6800:4009:811::200e: time=165ms

Ping statistics for 2404:6800:4009:811::200e:
    Packets: Sent = 8, Received = 8, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 89ms, Maximum = 165ms, Average = 120ms

C:\Windows\system32>
```

c. Ping your localhost. Explain what is the purpose.

```
C:\Windows\system32>ping localhost

Pinging LAPTOP-VI1R00K4 [::1] with 32 bytes of data:
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 = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Windows\system32>
```

The purpose of pinging the localhost (127.0.0.1) is to verify that your TCP/IP software is installed, started, and working properly.

2. Read the Unix manual page for traceroute OR help for tracert. Experiment with the various options. Describe the three things that you found most useful in the result. Answer the following question:

tracert is an important command useful for troubleshooting large networks and it is handy to find the number of hops from source to destination. tracert command also uses ICMP protocols to reach the destination. tracert follows a mechanism called time-to-live which prevents loop back of packets being sent continuously to a router. Every time a packet is being forwarded by a router, it is being decreased by 1 so eventually when it is zero, the network will discard the IP packet. In the various options given by tracert, the three options I've found most useful are -d, -w, -4 or -6. -d is used to avoid resolving the IP addresses to hostnames. -w is used to specify the wait timeout milliseconds for each reply. -4 or -6 to force IPv4 or IPv6 tracerouting.

a. Try tracert on google.com.

```
C:\Windows\system32>tracert google.com

Tracing route to google.com [2404:6800:4009:811::200e]
over a maximum of 30 hops:

  1    2 ms    3 ms    2 ms  2409:4072:6e1c:1abf::f7
  2    *      *      *      Request timed out.
  3   122 ms   391 ms   52 ms  2405:200:369:eeee:20::282
  4   113 ms   57 ms   41 ms  2405:200:801:2300::51a
  5    *      *      *      Request timed out.
  6    *      *      *      Request timed out.
  7   357 ms   680 ms   515 ms 2001:4860:1:1::d10
  8    70 ms    *      67 ms 2001:4860:1:1::d10
  9    67 ms    60 ms   41 ms 2404:6800:80f8::1
 10   239 ms    74 ms   58 ms 2001:4860:0:1::1108
 11   217 ms   272 ms  239 ms 2001:4860:0:e00::2
 12   339 ms    94 ms   264 ms 2001:4860:9:4001:7734
 13    *      *      *      Request timed out.
 14   210 ms   179 ms  147 ms 2001:4860:0:1::43d7
 15   106 ms    75 ms   116 ms bom12s06-in-x0e.1e100.net [2404:6800:4009:811::200e]

Trace complete.
```

b. Type tracert -d google.com

i. How many hops is your machine away from google.com?
www.google.comserver is 12 hops away from my machine.

```
C:\Windows\system32>tracert -d google.com
```

```
Tracing route to google.com [2404:6800:4009:811::200e]  
over a maximum of 30 hops:
```

1	20 ms	9 ms	3 ms	2409:4072:6e1c:1abf::f7
2	*	*	*	Request timed out.
3	105 ms	60 ms	50 ms	2405:200:369:eeee:20::282
4	122 ms	79 ms	111 ms	2405:200:801:2300::51a
5	*	*	*	Request timed out.
6	*	*	*	Request timed out.
7	*	*	51 ms	2001:4860:1:1::d10
8	70 ms	61 ms	*	2001:4860:1:1::d10
9	104 ms	73 ms	58 ms	2404:6800:80f8::1
10	80 ms	40 ms	47 ms	2001:4860:0:1::1108
11	68 ms	46 ms	60 ms	2001:4860:0:e00::2
12	130 ms	81 ms	123 ms	2001:4860::9:4001:7734
13	*	*	*	Request timed out.
14	102 ms	102 ms	92 ms	2001:4860:0:1::43d7
15	96 ms	84 ms	84 ms	2404:6800:4009:811::200e

```
Trace complete.
```

```
C:\Windows\system32>_
```

ii. Wait for a while and execute the same command again. Is the output the same as the first time? Observe and compare the difference, and explain the reason.

```
C:\Windows\system32>tracert -d google.com

Tracing route to google.com [2404:6800:4009:811::200e]
over a maximum of 30 hops:

  1    20 ms    9 ms    3 ms  2409:4072:6e1c:1abf::f7
  2    *        *        *      Request timed out.
  3   105 ms    60 ms    50 ms  2405:200:369:eeee:20::282
  4   122 ms    79 ms   111 ms  2405:200:801:2300::51a
  5    *        *        *      Request timed out.
  6    *        *        *      Request timed out.
  7    *        *        51 ms  2001:4860:1:1::d10
  8   70 ms    61 ms    *      2001:4860:1:1::d10
  9   104 ms    73 ms    58 ms  2404:6800:80f8::1
 10   80 ms    40 ms    47 ms  2001:4860:0:1::1108
 11   68 ms    46 ms    60 ms  2001:4860:0:e00::2
 12  130 ms    81 ms   123 ms  2001:4860::9:4001:7734
 13    *        *        *      Request timed out.
 14  102 ms   102 ms    92 ms  2001:4860:0:1::43d7
 15   96 ms    84 ms    84 ms  2404:6800:4009:811::200e

Trace complete.

C:\Windows\system32>
```

3.You have to read about NETSTAT from manual page or help before answering the below questions:

Netstat is used for checking the network statistics. It displays the types of services that run on ports. It displays the protocol used, private addresses, foreign addresses and the state of the connections.

```
C:\Windows\system32>netstat -r
=====
Interface List
19...08 8f c3 16 9c a5 .....Realtek PCIe GbE Family Controller
12...76 4c a1 76 ff af .....Microsoft Wi-Fi Direct Virtual Adapter
18...76 4c a1 76 ff bf .....Microsoft Wi-Fi Direct Virtual Adapter #2
4...74 4c a1 76 ff df .....MediaTek Wi-Fi 6 MT7921 Wireless LAN Card
10...74 4c a1 76 ff e0 .....Bluetooth Device (Personal Area Network) #3
1.....Software Loopback Interface 1
=====

IPv4 Route Table
=====
Active Routes:
Network Destination        Netmask          Gateway           Interface        Metric
0.0.0.0                    0.0.0.0          192.168.121.181   192.168.121.182   55
127.0.0.0                  255.0.0.0        On-link          127.0.0.1         331
127.0.0.1                  255.255.255.255  On-link          127.0.0.1         331
127.255.255.255            255.255.255.255  On-link          127.0.0.1         331
192.168.121.0               255.255.255.0    On-link          192.168.121.182   311
192.168.121.182             255.255.255.255  On-link          192.168.121.182   311
192.168.121.255             255.255.255.255  On-link          192.168.121.182   311
224.0.0.0                  240.0.0.0        On-link          127.0.0.1         331
224.0.0.0                  240.0.0.0        On-link          192.168.121.182   311
255.255.255.255            255.255.255.255  On-link          127.0.0.1         331
255.255.255.255            255.255.255.255  On-link          192.168.121.182   311
=====
Persistent Routes:
None

IPv6 Route Table
=====
Active Routes:
If Metric Network Destination      Gateway
4       71 ::/0 fe80::c804:2ff:fe8d:fe0b
1       331 ::1/128 On-link
4       71 2409:4072:6e1c:1abf::/64 On-link
4       311 2409:4072:6e1c:1abf:619d:4ea5:7175:93a1/128 On-link
4       311 2409:4072:6e1c:1abf:d938:e8e3:c6e9:c053/128 On-link
4       311 fe80::/64 On-link
4       311 fe80::d938:e8e3:c6e9:c053/128 On-link
1       331 ff00::/8 On-link
4       311 ff00::/8 On-link
=====
Persistent Routes:
None
```

ii. Use netstat to display about ethernet statistics.

```
C:\Windows\system32>netstat -e
Interface Statistics


```

	Received	Sent
Bytes	697990384	48570784
Unicast packets	593256	269480
Non-unicast packets	40	2616
Discards	0	0
Errors	0	0
Unknown protocols	0	

```
C:\Windows\system32>
```

4. What is the purpose of NSLOOKUP? Answer the following questions below:

nslookup is a network administration command-line tool for querying the Domain Name System to obtain the mapping between domain name and IP address, or other DNS records.

i. Use nslookup to find out the internet address of the domain amrita.edu.

```
C:\Windows\system32>nslookup -type=ns amrita.edu
Server: prithvi.amritanet.edu
Address: 172.17.18.2

Non-authoritative answer:
amrita.edu      nameserver = ns1.amrita.edu
amrita.edu      nameserver = ns3.amrita.edu
amrita.edu      nameserver = ns4.amrita.edu
amrita.edu      nameserver = ns2.amrita.edu

ns1.amrita.edu  internet address = 10.10.10.4
ns3.amrita.edu  internet address = 103.10.24.200
ns4.amrita.edu  internet address = 10.10.10.4
ns2.amrita.edu  internet address = 117.193.77.232
```

ii. What is the mail exchanger for the domain google.com?

```
C:\Windows\system32>nslookup -type=mx google.com
Server: UnKnown
Address: 192.168.121.181

Non-authoritative answer:
google.com      MX preference = 10, mail exchanger = smtp.google.com
C:\Windows\system32>
```

iii. What is the name server for amrita.edu?

```
C:\Windows\system32>nslookup -type=ns amrita.edu
Server: prithvi.amritanet.edu
Address: 172.17.18.2

Non-authoritative answer:
amrita.edu      nameserver = ns1.amrita.edu
amrita.edu      nameserver = ns3.amrita.edu
amrita.edu      nameserver = ns4.amrita.edu
amrita.edu      nameserver = ns2.amrita.edu
```

5.What is ARP and RARP? Answer the following questions below:

Address Resolution Protocol (ARP) is a communication protocol used to find the MAC (Media Access Control) address of a device from its IP address. This protocol is used when a device wants to communicate with another device on a Local Area Network or Ethernet. Reverse Address Resolution Protocol (RARP) is a networking protocol used by the client system in a local area network (LAN) to request its IPv4 address from the ARP gateway router table. A table is created by the network administrator in the gateway-router that is used to find out the MAC address to the corresponding IP address.

i . Use arp command to find the gateway address and host systems hardware address.


```

C:\Windows\system32>arp -a

Interface: 192.168.121.182 --- 0x4
    Internet Address      Physical Address      Type
    192.168.121.181      ca-04-02-8d-fe-0b    dynamic
    224.0.0.22           01-00-5e-00-00-16    static
    224.0.0.251          01-00-5e-00-00-fb    static
    224.0.0.252          01-00-5e-00-00-fc    static
    239.255.255.250      01-00-5e-7f-ff-fa    static
    255.255.255.255      ff-ff-ff-ff-ff-ff    static

C:\Windows\system32>

```

ii. How do you find the arp entries for particular interface?

To find the arp entries for a particular interface we need to use the **-N** flag along with the ip address.

c. How do delete an arp entry?

To delete an arp entry, we need to use the **-d flag** along with the ip address . To delete all the entries we need to use the wildcard flag(*) .

d. How do you add an arp entry in arpcache?

To add an arp entry we need to use **-s** flag along with IP address and MAC address.

EXAMPLE - arp -s 192.168.43.160 00-aa-00-62-c6-09

```
(kali㉿kali)-[~]
$ sudo tcpdump -i eth0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
22:22:45.713973 IP 10.0.2.5.bootpc > 10.0.2.3.bootps: BOOTP/DHCP, Request from 08:00:27:24:58:d0 (oui Unknown), length 282
22:22:45.729771 IP 10.0.2.3.bootps > 10.0.2.5.bootpc: BOOTP/DHCP, Reply, length 548
22:22:45.850531 IP 10.0.2.5.38626 > prithvi.amritanet.edu.domain: 49423+ PTR? 3.2.0.10.in-addr.arpa. (39)
22:22:45.852596 IP prithvi.amritanet.edu.domain > 10.0.2.5.38626: 49423 NXDomain 0/1/0 (74)
22:22:45.852862 IP 10.0.2.5.47386 > prithvi.amritanet.edu.domain: 39404+ PTR? 5.2.0.10.in-addr.arpa. (39)
22:22:45.855142 IP prithvi.amritanet.edu.domain > 10.0.2.5.47386: 39404 NXDomain 0/1/0 (74)
22:22:45.966524 IP 10.0.2.5.57749 > prithvi.amritanet.edu.domain: 17538+ PTR? 2.18.17.172.in-addr.arpa. (42)
22:22:50.770155 ARP, Request who-has 10.0.2.3 tell 10.0.2.5, length 28
22:22:50.770592 ARP, Reply 10.0.2.3 is-at 08:00:27:71:0d:10 (oui Unknown), length 46
22:22:50.981171 IP 10.0.2.5.50519 > varuna.amritanet.edu.domain: 17538+ PTR? 2.18.17.172.in-addr.arpa. (42)
22:22:50.986716 IP varuna.amritanet.edu.domain > 10.0.2.5.50519: 17538* 1/0/0 PTR prithvi.amritanet.edu. (77)
22:22:50.989042 IP 10.0.2.5.44346 > prithvi.amritanet.edu.domain: 53767+ PTR? 4.18.17.172.in-addr.arpa. (42)
22:22:50.991630 IP prithvi.amritanet.edu.domain > 10.0.2.5.44346: 53767* 1/0/0 PTR varuna.amritanet.edu. (76)
22:22:51.022064 ARP, Request who-has 10.0.2.1 tell 10.0.2.5, length 28
22:22:51.022395 ARP, Reply 10.0.2.1 is-at 52:54:00:12:35:00 (oui Unknown), length 46
22:22:51.139927 IP 10.0.2.5.46499 > prithvi.amritanet.edu.domain: 39023+ PTR? 1.2.0.10.in-addr.arpa. (39)
22:22:51.143994 IP prithvi.amritanet.edu.domain > 10.0.2.5.46499: 39023 NXDomain 0/1/0 (74)
```

7. Use Wireshark (Latest version) to solve the below scenarios:

1. You, as a SOC analyst noted that someone try to send information (PING) to unknown IP address and you are suspecting some malicious information might transferred in it. Analyze the log file.

a. Find the data transferred. – The data that is transferred in the packet is “pass!@#\$”

0000	00 0c 29 67 0b d2 74 c6 3b f2 eb db 08 00 45 00	..)g..t. ;.....E.
0010	00 24 34 f7 00 00 80 01 46 28 c0 a8 1f 10 c0 a8	.\$4..... F(.....
0020	1f 59 00 00 d7 c6 00 00 00 00 70 61 73 73 21 40	..Y..... ..pass!@
0030	23 24	#\$

b. Find the source and destination IP of that log.

Source Address: 192.168.31.16
Destination Address: 192.168.31.89
> Internet Control Message Protocol

0000	00 0c 29 67 0b d2 74 c6 3b f2 eb db 08 00 45 00	..)g..t. ;.....E.
0010	00 24 34 f7 00 00 80 01 46 28 c0 a8 1f 10 c0 a8	.\$4..... F(.....
0020	1f 59 00 00 d7 c6 00 00 00 00 70 61 73 73 21 40	..Y..... ..pass!@
0030	23 24	#\$

Source IP = 192.168.31.89, Destination IP = 192.168.31.16

c. Find the Data length (Bytes) and verify the checksum status on destination.

```

Internet Protocol Version 4, Src: 192.168.31.16, Dst: 192.168.31.89
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 36
  Identification: 0x34f7 (13559)
> 000. .... = Flags: 0x0
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 128
  Protocol: ICMP (1)
  Header Checksum: 0x4628 [validation disabled]
  [Header checksum status: Unverified]

```

2. Now you have found that some kind of file is been downloaded by insider in unencrypted web traffic. Your task is to

Length	Info
209	GET <u>/1.jpg HTTP/1.1</u>
22234	HTTP/1.1 200 OK (<u>JPEG JFIF image</u>)

a. Find the name and type of file.

NAME = 1.jpg , Type of file = JPEG JFIF

b. Export that file from that web traffic, then analyze the file for any secret information.



c. Find the hostname in which the file is stored. – 192.168.31.113

Destination	Dst code	Protocol
192.168.31.67	80	HTTP
192.168.31.113	59380	HTTP

3. Based upon their activities, auditing team has started investigation against them and found that the insider passed some sensitive information via call to someone. The traffic is being captured.

Sensitive Information: Password is “limbo”

No.	Time	Source	Src code	Destination	Dst code	Protocol	Length	Info
12692	2017/204	11:25:47.413904	192.168.31.8	5060	192.168.31.78	57332 SIP/SIP	325	Request: INVITE sip:1001@192.168.31.78:57332;instance=f3bc219541e
12703	2017/204	11:25:47.497561	192.168.31.78	57332	192.168.31.8	5060 SIP	131	Status: 100 Trying
12704	2017/204	11:25:47.497664	192.168.31.78	57332	192.168.31.8	5060 SIP	477	Status: 180 Ringing
13059	2017/204	11:25:49.433752	192.168.31.78	57332	192.168.31.8	5060 SIP	805	Status: 200 OK (INVITE)

- ```
> Via: SIP/2.0/UDP 192.168.31.8:5060;branch=z9hG4bK30e63862
> Contact: <sip:1001@192.168.31.78:57332>
> To: <sip:1001@192.168.31.78:57332;rinstance=fc3bc219541e9861;transport=UDP>
> From: "1002" <sip:1002@192.168.31.8>;tag=as1d95fb93
Call-ID: 01caab9b53b12efe00d3493a67ff695d@192.168.31.8:5060
[Generated Call-ID: 01caab9b53b12efe00d3493a67ff695d@192.168.31.8:5060]
> CSeq: 102 INVITE
User-Agent: Z 3.15.40006 rv2.8.20
Allow: Events: presence, km1, talk
```

**4. On further investigation, you have a suspect on some wireless device communications. List out the Bluetooth devices communications from this traffic and find the details about native Bluetooth adapter.**

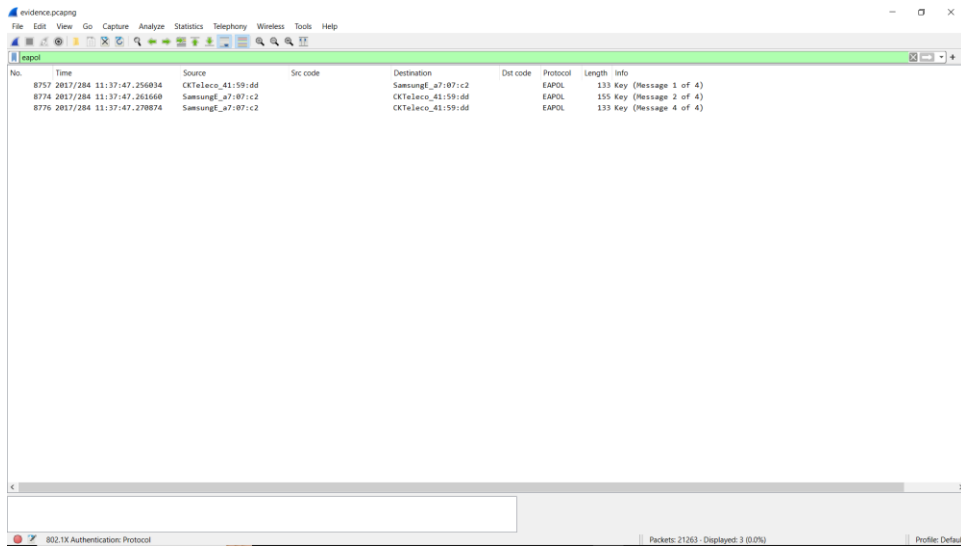
List of Bluetooth Devices:

| Bluetooth Devices |          |                           |             |                |                |             |              |                  |
|-------------------|----------|---------------------------|-------------|----------------|----------------|-------------|--------------|------------------|
| BD_ADDR           | OUI      | Name                      | LMP Version | LMP Subversion | Manufacturer   | HCI Version | HCI Revision | Is Local Adapter |
| 00:00:00:00:00:00 | 00:00:00 |                           |             |                |                |             |              |                  |
| 30:21:88:70:9c:18 |          | ZEB-INFINITY V2           | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |
| 30:22:00:33:ff:2b |          | KETTLE                    | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |
| 3cbb:fd:a7:07:c1  | SamsungE | Galaxy On5                | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |
| 4cbb:58:43:35:be  | ChicomE  | Virtual Bluetooth Adapter | 2.1 + EDR   | 256            | Unknown 0x%04x | 2.1 + EDR   | 256          | true             |
| a0:21:95:87:4d:7d | SamsungE | Vinayakar thunai          | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |
| a0:32:99:3c:65:52 | LenovoBe | Lenovo VIBE X3            | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |
| dce8:38:3e:54:6d  | CKTeleco | LS-4505                   | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |
| fc:58:fa:28:0d:c2 | ShenZhen | HP S6500                  | 2.1 + EDR   | 256            | Unknown 0x%04x |             |              |                  |

Bluetooth Adapter:

| Bluetooth Device - 4cbb:58:43:35:be (Virtual Bluetooth Adapter) |                           |         |
|-----------------------------------------------------------------|---------------------------|---------|
|                                                                 | Value                     | Changes |
| BD_ADDR                                                         | 4cbb:58:43:35:be          | 1       |
| OUI                                                             | ChicomE                   | 1       |
| Name                                                            | Virtual Bluetooth Adapter | 1       |
| Class of Device                                                 | 000100                    | 1       |
| LMP Version                                                     | 2.1 + EDR                 | 1       |
| LMP Subversion                                                  | 256                       | 1       |
| Manufacturer                                                    | Unknown 0x%04x            | 1       |
| HCI Version                                                     | 2.1 + EDR                 | 1       |
| HCI Revision                                                    | 256                       | 1       |
| Scan                                                            |                           |         |
| Authentication                                                  |                           |         |
| Encryption                                                      |                           |         |
| ACL MTU                                                         | 8192                      | 1       |
| ACL Total Packets                                               | 128                       | 1       |
| SCO MTU                                                         | 64                        | 1       |
| SCO Total Packets                                               | 128                       | 1       |
| LE ACL MTU                                                      |                           |         |
| LE ACL Total Packets                                            |                           |         |
| LE ISO MTU                                                      |                           |         |
| LE ISO Total Packets                                            |                           |         |
| Inquiry Mode                                                    | Results With RSSI         | 1       |
| Page Timeout                                                    |                           |         |
| 16 changes                                                      |                           |         |
| Close                                                           |                           |         |

**a. Analyze the captured WPA handshake from this traffic and report in detail about it to your administrator.**



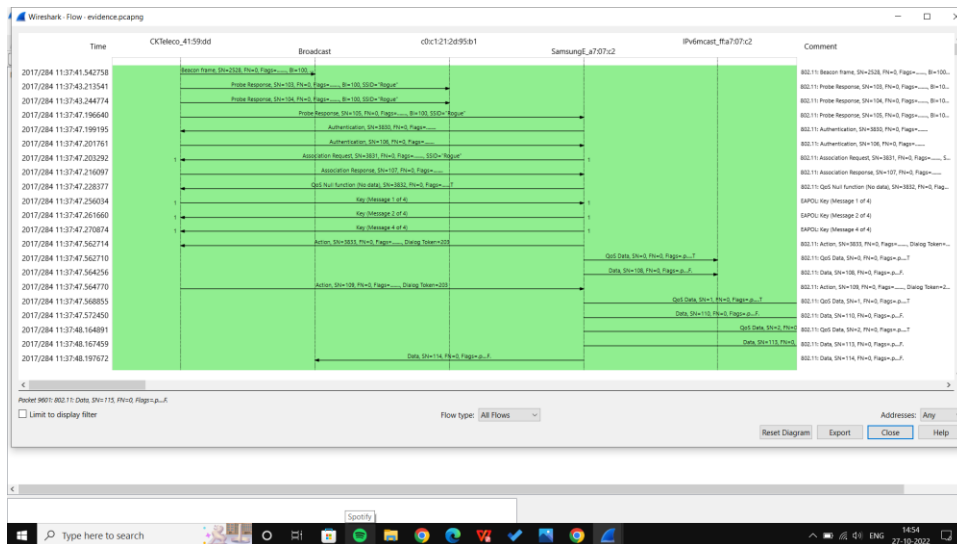
| No.  | Time                     | Source            | Src code | Destination       | Dest code | Protocol | Length | Info                 |
|------|--------------------------|-------------------|----------|-------------------|-----------|----------|--------|----------------------|
| 8757 | 2017/284 11:37:47.256034 | CKTeleco_41:59:dd |          | SamsungF_a7:07:c2 |           | EAPOL    | 133    | Key (Message 1 of 4) |
| 8774 | 2017/284 11:37:47.261668 | SamsungF_a7:07:c2 |          | CKTeleco_41:59:dd |           | EAPOL    | 155    | Key (Message 2 of 4) |
| 8776 | 2017/284 11:37:47.270874 | SamsungF_a7:07:c2 |          | CKTeleco_41:59:dd |           | EAPOL    | 133    | Key (Message 4 of 4) |

Out of 4 only 3 is available

**b. Geo locate all the endpoint of wireless devices.**

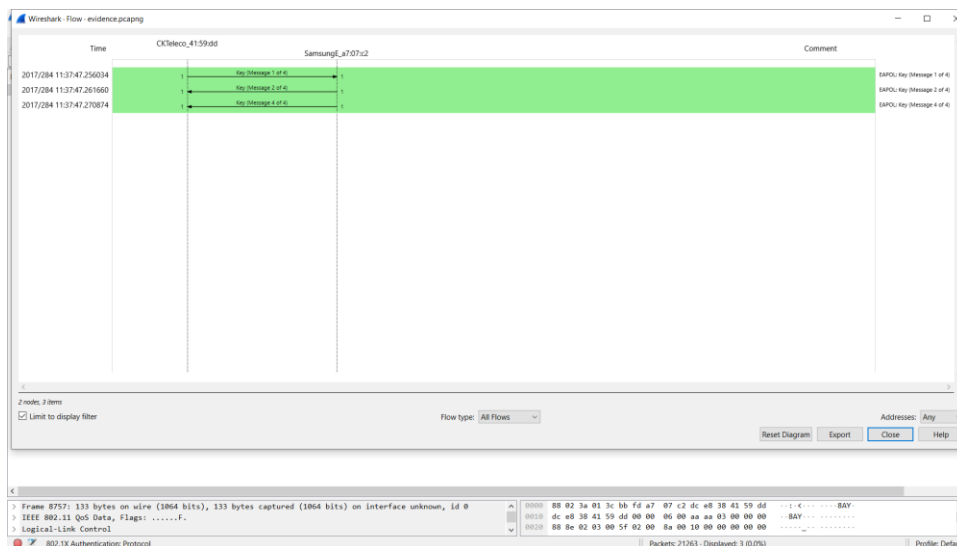
## c. Analyze the protocol level information transfer between wireless devices.

### Go to statistics and go to flow graph



Left hand side-time stamp

Limit to Display filter then only we get particular display





| No.  | Time                     | Source            | Src code | Destination              | Dst code | Protocol | Length | Info                                                            |
|------|--------------------------|-------------------|----------|--------------------------|----------|----------|--------|-----------------------------------------------------------------|
| 24   | 2017/284 11:37:41.542758 | CKTeleco_41:59:dd |          | Broadcast                |          | 802.11   | 176    | Beacon frame, SN=2528, FH=0, Flags=....., BI=100, SSID="Rogue"  |
| 3121 | 2017/284 11:37:43.213541 | CKTeleco_41:59:dd |          | c0:c1:21:2d:95:b1        |          | 802.11   | 165    | Probe Response, SN=103, FH=0, Flags=....., BI=100, SSID="Rogue" |
| 3162 | 2017/284 11:37:43.244774 | CKTeleco_41:59:dd |          | c0:c1:21:2d:95:b1        |          | 802.11   | 165    | Probe Response, SN=104, FH=0, Flags=....., BI=100, SSID="Rogue" |
| 8674 | 2017/284 11:37:47.196640 | CKTeleco_41:59:dd |          | Samsung_a7:07:c2         |          | 802.11   | 165    | Probe Response, SN=105, FH=0, Flags=....., BI=100, SSID="Rogue" |
| 8675 | 2017/284 11:37:47.196634 |                   |          | CKTeleco_41:59:dd (dc... |          | 802.11   | 10     | Acknowledgement, Flags=.....                                    |
| 8679 | 2017/284 11:37:47.199195 | Samsung_a7:07:c2  |          | CKTeleco_41:59:dd        |          | 802.11   | 41     | Authentication, SN=3830, FH=0, Flags=.....                      |
| 8693 | 2017/284 11:37:47.201761 | CKTeleco_41:59:dd |          | Samsung_a7:07:c2         |          | 802.11   | 30     | Authentication, SN=106, FH=0, Flags=.....                       |
| 8694 | 2017/284 11:37:47.202267 |                   |          | CKTeleco_41:59:dd (dc... |          | 802.11   | 10     | Acknowledgement, Flags=.....                                    |
| 8696 | 2017/284 11:37:47.203292 | Samsung_a7:07:c2  |          | CKTeleco_41:59:dd        |          | 802.11   | 129    | Association Request, SN=3831, FH=0, Flags=....., SSID="Rogue"   |
| 8698 | 2017/284 11:37:47.216097 | CKTeleco_41:59:dd |          | Samsung_a7:07:c2         |          | 802.11   | 124    | Association Response, SN=107, FH=0, Flags=.....                 |
| 8699 | 2017/284 11:37:47.216604 |                   |          | CKTeleco_41:59:dd (dc... |          | 802.11   | 10     | Acknowledgement, Flags=.....                                    |
| 8707 | 2017/284 11:37:47.228377 | Samsung_a7:07:c2  |          | CKTeleco_41:59:dd        |          | 802.11   | 26     | QoS Null function (No data), SN=3832, FH=0, Flags=.....T        |
| 8757 | 2017/284 11:37:47.256834 | CKTeleco_41:59:dd |          | Samsung_a7:07:c2         |          | EAPOL    | 133    | key (Message 1 of 4)                                            |
| 8758 | 2017/284 11:37:47.256829 |                   |          | CKTeleco_41:59:dd (dc... |          | 802.11   | 10     | Acknowledgement, Flags=.....                                    |

## Analysis of protocol information transfer:

- AP Beacon, i.e., announces presence and capabilities of AP.
- Probe Request packet, i.e., client looking for AP.
- Probe Response packet, i.e., AP responding to client.
- Open-authentication System packets, i.e., client sending authentication.
- 4-Way Handshake.