COMPUTER NETWORK SECURITY LAB -02

Name: Pavan R Kashyap SRN: PES1UG20CS280

5th Semester E section

1. **Task 2.1A- Understanding how a Sniffer Works**

   Host A is sending ICMP messages/ packets to the IP 10.9.0.1. When the sniff program is executed on the Attacker system, the program sniffs ICMP packets being sent in the network. The source IP, destination IP and protocols of the packets(ICMP is the filter used so protocol will be ICMP only) are displayed on the attacker's screen.

```
PES1UG20CS280_ROOT(10.0.2.5) -$./sniff
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.9.0.1
   Protocol: ICMP
```

Question 1-

pcap_open_live() system call must be used to create a handler that sniffs packets in the network. It also turns on the promiscuous mode.

pcap_compile() is used to compile the filter that is to be used. The result of the compilation is stored in a handler.

pcap_setfilter() is used to specify a filter program.

pcap_loop(callback fn.) is used to process packets from a live capture. Every time a packet is captured, the callback function is called.

To summarise, the ethernet interface must be setup, PCAP must be initialized, filters must be compiled, the sniff code must be executed and finally terminated after its role is done.

<u>**COMPUTER NETWORK SECURITY LAB -02**</u>

Name: Pavan R Kashyap SRN: PES1UG20CS280

Question 2

Whenever network interfaces need to be accessed, root access is required. The program uses raw sockets to send/receive packets in the network. Without the root privileges, the NIC card would not be accessible and hence we will not be able to use the raw sockets.

When the su seed command is used, the root user switches to the seed user. The seed user does not have/hold root privileges. When the sniff program is executed, the program is trying to access certain resources (raw sockets in this case) that it is not permitted to use/access. Hence, Segmentation Fault is displayed.

```
PES1UG20CS280(10.0.2.5) -$./sniff
Segmentation fault
PES1UG20CS280(10.0.2.5) -$
```

Question 3

When the promiscuous mode is on, we see the sniffer code running and displaying all the sniffed packets in the network (as seen in previous page). When the promiscuous mode is switched off, the host system drops packets that are not meant for it and therefore, no sniffed packet information is displayed on screen when program is executed.

```
PES1UG20CS280_ROOT(10.0.2.5) -$./sniff
```

When 10.9.0.6 is pinged on Host A's system.

```
PING 10.9.0.6 (10.9.0.6) 56(84) bytes of data.
```

COMPUTER NETWORK SECURITY LAB -02

Name: Pavan R Kashyap SRN: PES1UG20CS280

2. Task 2.1B-ICMP

a) Capture the ICMP packets between two specific hosts

The two hosts in this case are 10.0.2.4 and 10.0.2.5.
When ping command is executed on Host A(10.0.2.4)'s system, ICMP packets are sent b/w Host A and server(10.0.2.5). Host A sends Echo Request messages to 10.0.2.5 and 10.0.2.5 replies back with Echo Response messages.
The attacker sniffs these packets in the network and displays details about the packets.

```
PES1UG20CS280_ROOT(10.0.2.5) -$./sniff
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: ICMP
       From: 10.0.2.5
         To: 10.0.2.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: ICMP
       From: 10.0.2.5
         To: 10.0.2.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: ICMP
       From: 10.0.2.5
         To: 10.0.2.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 10.0.2.5
```

The corresponding Wireshark output is attached below-

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-08-31 10:19:12.2098628…	::1	::1	UDP	64	53720 → 56105 Len=0
2	2022-08-31 10:19:32.2188459…	::1	::1	UDP	64	53720 → 56105 Len=0
3	2022-08-31 10:19:41.7784075…	PcsCompu_c6:fa:69		ARP	44	Who has 10.0.2.5? Tell 10.0.2.4
4	2022-08-31 10:19:41.7790200…	PcsCompu_94:43:70		ARP	62	10.0.2.5 is at 08:00:27:94:43:70
5	2022-08-31 10:19:41.7790336…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=1/256, ttl=64 (reply in 6)
6	2022-08-31 10:19:41.7792728…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=1/256, ttl=64 (request in 5)
7	2022-08-31 10:19:42.7815069…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=2/512, ttl=64 (reply in 8)
8	2022-08-31 10:19:42.7819948…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=2/512, ttl=64 (request in 7)
9	2022-08-31 10:19:43.8109095…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=3/768, ttl=64 (reply in 10)
10	2022-08-31 10:19:43.8115001…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=3/768, ttl=64 (request in 9)
11	2022-08-31 10:19:43.8118018…	::1	::1	UDP	64	53720 → 56105 Len=0
12	2022-08-31 10:19:44.8329935…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=4/1024, ttl=64 (reply in 13)
13	2022-08-31 10:19:44.8338323…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=4/1024, ttl=64 (request in 12)
14	2022-08-31 10:19:45.8367921…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=5/1280, ttl=64 (reply in 15)
15	2022-08-31 10:19:45.8373352…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=5/1280, ttl=64 (request in 14)
16	2022-08-31 10:19:46.8508035…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=6/1536, ttl=64 (reply in 17)
17	2022-08-31 10:19:46.8513736…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=6/1536, ttl=64 (request in 16)
18	2022-08-31 10:19:46.9178880…	PcsCompu_94:43:70		ARP	62	Who has 10.0.2.4? Tell 10.0.2.5
19	2022-08-31 10:19:46.9179278…	PcsCompu_c6:fa:69		ARP	44	10.0.2.4 is at 08:00:27:c6:fa:69
20	2022-08-31 10:19:47.8772007…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=7/1792, ttl=64 (reply in 21)
21	2022-08-31 10:19:47.8794861…	10.0.2.5	10.0.2.4	ICMP	100	Echo (ping) reply id=0x0d1c, seq=7/1792, ttl=64 (request in 20)
22	2022-08-31 10:19:48.8790788…	10.0.2.4	10.0.2.5	ICMP	100	Echo (ping) request id=0x0d1c, seq=8/2048, ttl=64 (reply in 23)

COMPUTER NETWORK SECURITY LAB -02

Name: Pavan R Kashyap SRN: PES1UG20CS280

b) Capture the TCP packets that have a destination port range from to sort 10 - 100.

telnet 10.0.2.5 command is used here.

TCP packets are exchanged b/w Host A(10.0.2.4) and server(10.0.2.5). The sniffer program sniffs the TCP packets being exchanged in the network and displays them on screen.

```
PES1UG20CS280_ROOT(10.0.2.5) -$./sniff
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
       From: 10.0.2.4
         To: 10.0.2.5
   Protocol: TCP
```

On Wireshark, the TCP and TELNET packets being exchanged are displayed-

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-08-31 10:50:48.2580774…	10.0.2.5	224.0.0.251	MDNS	185	Standard query 0x0000 PTR _nfs._tcp.local, "QM" question PTR _ipp._tcp.local, "QM" question PTR _ipps
2	2022-08-31 10:50:49.4846790…	fe80::7cf3:ab06:1a8…	ff02::fb	MDNS	205	Standard query 0x0000 PTR _nfs._tcp.local, "QM" question PTR _ipp._tcp.local, "QM" question PTR _ipps
3	2022-08-31 10:50:52.6705796…	::1	::1	UDP	64	53720 → 56105 Len=0
4	2022-08-31 10:51:12.6851488…	::1	::1	UDP	64	53720 → 56105 Len=0
5	2022-08-31 10:51:15.2643525…	10.0.2.4	10.0.2.5	TCP	76	54706 → 23 [SYN] Seq=3288575266 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1214327 TSecr=0 WS=128
6	2022-08-31 10:51:15.2646525…	10.0.2.5	10.0.2.4	TCP	76	23 → 54706 [SYN, ACK] Seq=1493844712 Ack=3288575267 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=12156
7	2022-08-31 10:51:15.2646754…	10.0.2.4	10.0.2.5	TCP	68	54706 → 23 [ACK] Seq=3288575267 Ack=1493844713 Win=29312 Len=0 TSval=1214328 TSecr=1215639
8	2022-08-31 10:51:15.2648303…	10.0.2.4	10.0.2.5	TELNET	95	Telnet Data ...
9	2022-08-31 10:51:15.2649783…	10.0.2.5	10.0.2.4	TCP	68	23 → 54706 [ACK] Seq=1493844713 Ack=3288575294 Win=29056 Len=0 TSval=1215639 TSecr=1214328
10	2022-08-31 10:51:15.3022014…	RealtekU_12:35:00		ARP	62	Who has 10.0.2.5? Tell 10.0.2.1
11	2022-08-31 10:51:15.3028912…	10.0.2.5	10.0.2.4	TELNET	80	Telnet Data ...
12	2022-08-31 10:51:15.3029171…	10.0.2.4	10.0.2.5	TCP	68	54706 → 23 [ACK] Seq=3288575294 Ack=1493844725 Win=29312 Len=0 TSval=1214337 TSecr=1215649
13	2022-08-31 10:51:15.3031008…	10.0.2.5	10.0.2.4	TELNET	107	Telnet Data ...
14	2022-08-31 10:51:15.3031075…	10.0.2.4	10.0.2.5	TCP	68	54706 → 23 [ACK] Seq=3288575294 Ack=1493844764 Win=29312 Len=0 TSval=1214337 TSecr=1215649
15	2022-08-31 10:51:15.3032250…	10.0.2.4	10.0.2.5	TELNET	143	Telnet Data ...
16	2022-08-31 10:51:15.3034088…	10.0.2.5	10.0.2.4	TCP	68	23 → 54706 [ACK] Seq=1493844764 Ack=3288575369 Win=29056 Len=0 TSval=1215649 TSecr=1214337
17	2022-08-31 10:51:15.3036815…	10.0.2.5	10.0.2.4	TELNET	71	Telnet Data ...
18	2022-08-31 10:51:15.3037218…	10.0.2.4	10.0.2.5	TELNET	71	Telnet Data ...
19	2022-08-31 10:51:15.3045952…	10.0.2.5	10.0.2.4	TELNET	71	Telnet Data ...
20	2022-08-31 10:51:15.3046823…	10.0.2.4	10.0.2.5	TELNET	71	Telnet Data ...
21	2022-08-31 10:51:15.3048462…	10.0.2.5	10.0.2.4	TELNET	88	Telnet Data ...
22	2022-08-31 10:51:15.3449321…	10.0.2.4	10.0.2.5	TCP	68	54706 → 23 [ACK] Seq=3288575375 Ack=1493844790 Win=29312 Len=0 TSval=1214348 TSecr=1215649

▸ Frame 22: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0

3. **Task 2.1C –Sniffing Passwords**

telnet 10.0.2.5 used in this case.

When telnet command is used on Host A's terminal (10.0.2.4) , it asks the user to enter the login credentials. When the appropriate username and password is given, then Host A gets remote access to the system/server (10.0.2.5). The terminal line after telnet command is the terminal line of the system/server(10.0.2.5).

```
root@VM:/home/seed/Desktop/CNS/Code# telnet 10.0.2.5
Trying 10.0.2.5...
Connected to 10.0.2.5.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
Last login: Wed Aug 31 11:09:37 EDT 2022 from 10.0.2.5 on pts/20
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.

PES1UG20CS280(10.0.2.5) -$
```

The sniff program captures the username and password entered by user at Host A and displays that information to the attacker. The output below is what is displayed on the Attacker's system/terminal when the credentials are entered correctly at Host A's side.

```
Terminal
PES1UG20CS280_ROOT(10.0.2.5) -$./sniff
�����������  ��!��"��'����#������ ��#��'�������!��"���� ������#�����'�������
��������������Ubuntu 16.04.2 LTS
�VM login: ssee�ee�dd�
�Password: d�ee�s�
�Last login: Wed Aug 31 11:09:37 EDT 2022 from 10.0.2.5 on pts/20
��Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

1 package can be updated.
0 updates are security updates.

^Z
[5]+  Stopped                 ./sniff
PES1UG20CS280_ROOT(10.0.2.5) -$
```

COMPUTER NETWORK SECURITY LAB -02

Name: Pavan R Kashyap SRN: PES1UG20CS280

4. **TASK 2.2 -- Spoof an ICMP Echo Request**

   Spoofed ICMP packets are sent from the Attacker (Attacker uses a fake source I.P address) to a server/machine that is live.

   a) If destination is 8.8.8.8, then the Wireshark capture is as follows-

Apply a display filter ... <Ctrl-/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-08-31 11:16:55.4220461…	10.0.2.5	10.0.2.3	DHCP	342	DHCP Request - Transaction ID 0x3b7f7
2	2022-08-31 11:16:55.4290200…	10.0.2.3	10.0.2.5	DHCP	590	DHCP ACK - Transaction ID 0x3b7f7
3	2022-08-31 11:17:00.6419736…	PcsCompu_94:43:70	PcsCompu_2b:d0:bb	ARP	60	Who has 10.0.2.3? Tell 10.0.2.5
4	2022-08-31 11:17:00.6419903…	PcsCompu_2b:d0:bb	PcsCompu_94:43:70	ARP	60	10.0.2.3 is at 08:00:27:2b:d0:bb
5	2022-08-31 11:19:14.1375167…	1.2.3.4	8.8.8.8	ICMP	60	Echo (ping) request id=0x0000, seq=0/
6	2022-08-31 11:19:14.1607442…	RealtekU_12:35:00	Broadcast	ARP	60	Gratuitous ARP for 10.0.2.1 (Request)
7	2022-08-31 11:19:19.1408526…	PcsCompu_94:43:70	RealtekU_12:35:00	ARP	60	Who has 10.0.2.1? Tell 10.0.2.5
8	2022-08-31 11:19:19.1408615…	RealtekU_12:35:00	PcsCompu_94:43:70	ARP	60	10.0.2.1 is at 52:54:00:12:35:00

Echo request message is sent out to the server and that is captured by Wireshark. The equivalent ICMP response packet is not displayed on the screen as the response packet does not enter the local network.

   b) If destination is Host A's IP address, then the Wireshark capture is as follows-

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-08-31 11:22:38.3860655…	1.2.3.4	10.0.2.4	ICMP	60	Echo (ping) request id=0x0000, seq=0/0, ttl=20 (reply in 2)
2	2022-08-31 11:22:38.3861077…	10.0.2.4	1.2.3.4	ICMP	42	Echo (ping) reply id=0x0000, seq=0/0, ttl=64 (request in 1)
3	2022-08-31 11:22:43.4347771…	PcsCompu_94:43:70	PcsCompu_c6:fa:69	ARP	60	Who has 10.0.2.4? Tell 10.0.2.5
4	2022-08-31 11:22:43.4348012…	PcsCompu_c6:fa:69	PcsCompu_94:43:70	ARP	42	10.0.2.4 is at 08:00:27:c6:fa:69
5	2022-08-31 11:22:43.5530729…	PcsCompu_c6:fa:69	RealtekU_12:35:00	ARP	42	Who has 10.0.2.1? Tell 10.0.2.4
6	2022-08-31 11:22:43.5531949…	RealtekU_12:35:00	PcsCompu_c6:fa:69	ARP	60	10.0.2.1 is at 52:54:00:12:35:00

Echo request and response messages are being exchanged in the local network and hence both packets' information is displayed on screen.

Question 4

No, the checksum for the IP header is calculated and verified by the kernel of the Operating System only (it does not need to be manually calculated by the user).

Question 5

Raw sockets are restricted to root because if, otherwise it would break the other rules for networking that are in place. This is decided upon by the authorities who set the rules for networking. pcap_open_live() fails if root privilege is not given.

COMPUTER NETWORK SECURITY LAB -02

Name: Pavan R Kashyap SRN: PES1UG20CS280

5. TASK 2.3 –Sniffing and Spoofing

Attacker sniffs the network for any ICMP packets. Host A is pinging an arbitrary IP address that does not exist (it sends Echo request packets to this IP). The sniff program detects these ICMP packets and generates spoofed ICMP response packets that back to Host A. So, Host A is under the assumption that 1.2.3.4 is responding to its ping requests when actually it is the Attacker.

The ping output on Host A's terminal

```
PING 1.2.3.4 (1.2.3.4) 56(84) bytes of data.
64 bytes from 1.2.3.4: icmp_seq=1 ttl=20 time=378 ms
64 bytes from 1.2.3.4: icmp_seq=2 ttl=20 time=398 ms
64 bytes from 1.2.3.4: icmp_seq=1 ttl=20 time=2429 ms
64 bytes from 1.2.3.4: icmp_seq=3 ttl=20 time=422 ms
64 bytes from 1.2.3.4: icmp_seq=2 ttl=20 time=2445 ms
64 bytes from 1.2.3.4: icmp_seq=4 ttl=20 time=442 ms
64 bytes from 1.2.3.4: icmp_seq=1 ttl=20 time=4475 ms
64 bytes from 1.2.3.4: icmp_seq=3 ttl=20 time=2469 ms
64 bytes from 1.2.3.4: icmp_seq=5 ttl=20 time=465 ms
64 bytes from 1.2.3.4: icmp_seq=2 ttl=20 time=4493 ms
64 bytes from 1.2.3.4: icmp_seq=4 ttl=20 time=2490 ms
64 bytes from 1.2.3.4: icmp_seq=6 ttl=20 time=487 ms
^Z
[1]+  Stopped                 ping 1.2.3.4
root@VM:/home/seed/Desktop/CNS/Code#
```

The Wireshark display of the request response packets being exchanged in the network-

1	2022-08-31 11:28:14.3629837…	10.0.2.4	10.0.2.3	DHCP	342	DHCP Request - Transaction ID 0x7b16f94f
2	2022-08-31 11:28:14.3782133…	10.0.2.3	10.0.2.4	DHCP	590	DHCP ACK - Transaction ID 0x7b16f94f
3	2022-08-31 11:28:19.4265014…	PcsCompu_c6:fa:69	PcsCompu_2b:d0:bb	ARP	42	Who has 10.0.2.3? Tell 10.0.2.4
4	2022-08-31 11:28:19.4266672…	PcsCompu_2b:d0:bb	PcsCompu_c6:fa:69	ARP	60	10.0.2.3 is at 08:00:27:2b:d0:bb
5	2022-08-31 11:28:25.4474330…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) request id=0x13c6, seq=1/256, ttl=64 (reply in 6)
6	2022-08-31 11:28:25.8261979…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=1/256, ttl=20 (request in 5)
7	2022-08-31 11:28:26.4521942…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) request id=0x13c6, seq=2/512, ttl=64 (reply in 9)
8	2022-08-31 11:28:26.8504428…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=1/256, ttl=20
9	2022-08-31 11:28:26.8504559…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=2/512, ttl=20 (request in 7)
10	2022-08-31 11:28:27.4540243…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) request id=0x13c6, seq=3/768, ttl=64 (reply in 13)
11	2022-08-31 11:28:27.8764745…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=1/256, ttl=20
12	2022-08-31 11:28:27.8765085…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=2/512, ttl=20
13	2022-08-31 11:28:27.8765100…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=3/768, ttl=20 (request in 10)
14	2022-08-31 11:28:28.4559315…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) request id=0x13c6, seq=4/1024, ttl=64 (reply in 18)
15	2022-08-31 11:28:28.8979774…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=1/256, ttl=20
16	2022-08-31 11:28:28.8979910…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=2/512, ttl=20
17	2022-08-31 11:28:28.8980562…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=3/768, ttl=20
18	2022-08-31 11:28:28.8980590…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=4/1024, ttl=20 (request in 14)
19	2022-08-31 11:28:29.4577258…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) request id=0x13c6, seq=5/1280, ttl=64 (no response found!)
20	2022-08-31 11:28:29.9230752…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=1/256, ttl=20
21	2022-08-31 11:28:29.9231211…	10.0.2.4	1.2.3.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=2/512, ttl=20
22	2022-08-31 11:28:29.9232046…	1.2.3.4	10.0.2.4	ICMP	98	Echo (ping) reply id=0x13c6, seq=3/768, ttl=20

The program sniffs and displays the details (Source IP, Destination IP and Protocol) of the ICMP packets in the network.

```
PES1UG20CS280_ROOT(10.0.2.5) -$./sniffspoof
       From: 10.0.2.4
         To: 1.2.3.4
   Protocol: ICMP
       From: 1.2.3.4
         To: 10.0.2.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 1.2.3.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 1.2.3.4
   Protocol: ICMP
       From: 1.2.3.4
         To: 10.0.2.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 1.2.3.4
   Protocol: ICMP
       From: 1.2.3.4
         To: 10.0.2.4
   Protocol: ICMP
       From: 10.0.2.4
         To: 1.2.3.4
   Protocol: ICMP
       From: 1.2.3.4
         To: 10.0.2.4
   Protocol: ICMP
```