

Project 2: MITM & Pharming Attacks in Wi-Fi Networks

1. Man-In-The-Middle (MITM) Attack

Testing Environment:

Attacker: 192.168.0.177 (00:0c:29:15:0b:91)

Victim : 192.168.0.163 (00:0c:29:e0:ed:97)

Gateway: 192.168.0.1 (1c:3b:f3:bd:78:23) Tp-LinkT_bd

Both attacker and victim are Ubuntu VM provided by the requirement uses bridge mode connect to the real Wi-Fi router.

No.	Time	Source	Destination	Protocol	Length	Info
35	4.622881900	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=64 (no response f...
36	4.622920124	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=63 (reply in 37)
37	4.626775006	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=58 (request in 36)
38	4.626786246	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=57

▶ Frame 35: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						
▼ Ethernet II, Src: Vmware_e0:ed:97 (00:0c:29:e0:ed:97), Dst: Vmware_15:0b:91 (00:0c:29:15:0b:91)						
▶ Destination: Vmware_15:0b:91 (00:0c:29:15:0b:91)						
▶ Source: Vmware_e0:ed:97 (00:0c:29:e0:ed:97)						
Type: IPv4 (0x0800)						

Ping packet first sent from victim(e0:ed:97) to attacker(15:0b:91).

No.	Time	Source	Destination	Protocol	Length	Info
35	4.622881900	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=64 (no response f...
36	4.622920124	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=63 (reply in 37)
37	4.626775006	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=58 (request in 36)
38	4.626786246	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=57

▶ Frame 36: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						
▼ Ethernet II, Src: Vmware_15:0b:91 (00:0c:29:15:0b:91), Dst: Tp-LinkT_bd:78:23 (1c:3b:f3:bd:78:23)						
▶ Destination: Tp-LinkT_bd:78:23 (1c:3b:f3:bd:78:23)						
▶ Source: Vmware_15:0b:91 (00:0c:29:15:0b:91)						
Type: IPv4 (0x0800)						

And transmit to router from attacker(15:0b:91).

No.	Time	Source	Destination	Protocol	Length	Info
35	4.622881900	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=64 (no response f...
36	4.622920124	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=63 (reply in 37)
37	4.626775006	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=58 (request in 36)
38	4.626786246	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=57

▶ Frame 37: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						
▼ Ethernet II, Src: Tp-LinkT_bd:78:23 (1c:3b:f3:bd:78:23), Dst: Vmware_15:0b:91 (00:0c:29:15:0b:91)						
▶ Destination: Vmware_15:0b:91 (00:0c:29:15:0b:91)						
▶ Source: Tp-LinkT_bd:78:23 (1c:3b:f3:bd:78:23)						
Type: IPv4 (0x0800)						

The return packet also sent to the attacker(15:0b:91) first from the router.

No.	Time	Source	Destination	Protocol	Length	Info
35	4.622881900	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=64 (no response f...
36	4.622920124	192.168.0.163	8.8.8.8	ICMP	98	Echo (ping) request id=0x4cbf, seq=1/256, ttl=63 (reply in 37)
37	4.626775006	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=58 (request in 36)
38	4.626786246	8.8.8.8	192.168.0.163	ICMP	98	Echo (ping) reply id=0x4cbf, seq=1/256, ttl=57

▶ Frame 38: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						
▼ Ethernet II, Src: Vmware_15:0b:91 (00:0c:29:15:0b:91), Dst: Vmware_e0:ed:97 (00:0c:29:e0:ed:97)						
▶ Destination: Vmware_e0:ed:97 (00:0c:29:e0:ed:97)						
▶ Source: Vmware_15:0b:91 (00:0c:29:15:0b:91)						
Type: IPv4 (0x0800)						

Finally, send packet back to the victim(e0:ed:97) from the attacker(15:0b:91).

The overall path is “victim -> attacker -> router -> attacker -> victim”.

```

Available devices
-----
IP Address      MAC Address
-----
192.168.0.129   d8:bb:c1:9c:54:94
192.168.0.163   00:0c:29:e0:ed:97
192.168.0.177   2c:6d:c1:0f:2f:0d
192.168.0.127   c4:b3:01:c5:e9:3b
192.168.0.147   56:c4:1e:5b:7d:d2
192.168.0.158   14:7d:da:bf:91:d6
192.168.0.185   0e:60:51:fb:1c:a1
-----
Username: 123
Password: 999

```

Also, the program could also get the username and password from the e3 login page when the victim try to login.

2. Pharming Attack

The screenshot on the left was captured on victim(.163). First use the ‘dig’ command to see what we got from the DNS query and it shows that we get the attacker’s server. Uses ‘curl’ command and we got the spoofed web page.

```

cs2022@ubuntu:~$ sudo systemd-resolve --flush-caches
cs2022@ubuntu:~$ dig www.nycu.edu.tw

; <<> DiG 9.11.3-ubuntu1.14-Ubuntu <<> www.nycu.edu.tw
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 51466
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:: udp: 65494
;; QUESTION SECTION:
;www.nycu.edu.tw.                IN      A

;; ANSWER SECTION:
www.nycu.edu.tw.                0       IN      A      140.113.207.237

;; Query time: 12 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Thu Apr 14 14:54:22 CST 2022
;; MSG SIZE rcvd: 60

cs2022@ubuntu:~$ curl www.nycu.edu.tw
<!DOCTYPE html>
<html>
<body>

<h1 style="text-align:center;">This is a spoofed web page!</h1>
<p style="text-align:center;"><b>Good Job!</b></p>

</body>
</html>

```

3. Defend against ARP Spoofing Attack

• Devices

- Fixed ARP table — Not to change the ARP table when receive the arp response.

• Router

- Router could check the ARP table to find duplicate MAC address but different IP.
- Send the ARP packet to every devices periodically to ensure their gateway MAC address is correct.
- Devices could not send ARP packet to each other.