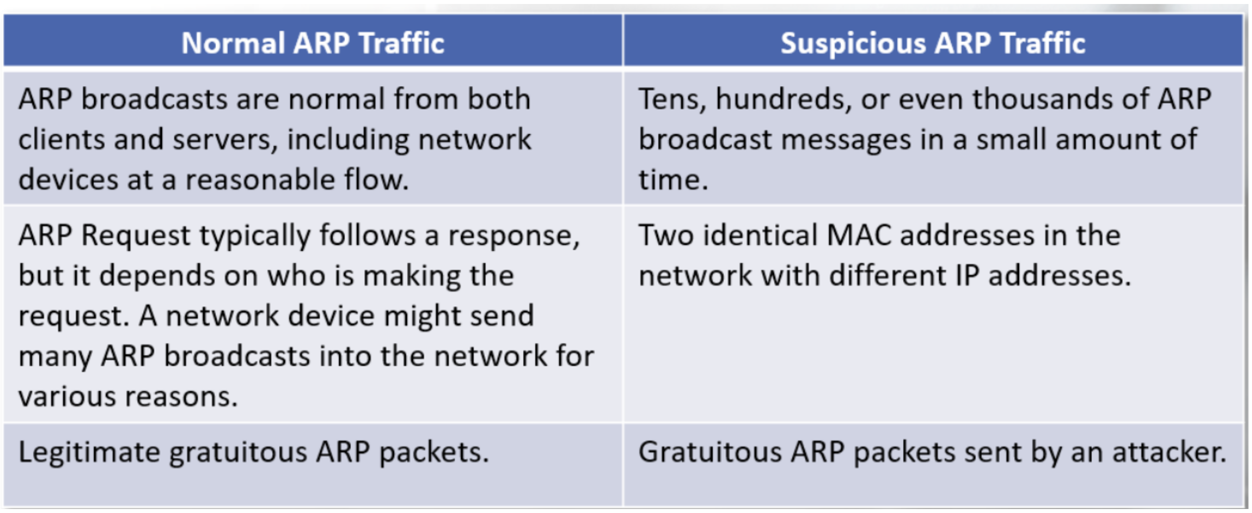
ARP (Address Resolution Protocol) Traffic:

ARP is Layer 2 protocol as it deal with MAC addresses, which it uses IP to map IP addresses to MAC addresses. Mean if we want communicate with anything within the network we need their MAC address, like if we want to communicate to a workstation in the network, we need to get their IP first which is easy to get and then we will use that IP to get their MAC address using ARP and then we will be able to communicate with each other.

ARP communicate using **REQUEST message** and **RESPONSE message**. In the header we can see if its Request or response by see these 2 things:

- Request (1): this mean the REQUEST in the ARP protocol Header.
- Reply (2): this mean the response when we see it in the ARP protocol header.

Here are some facts to distinguish normal and suspicious ARP Traffic:



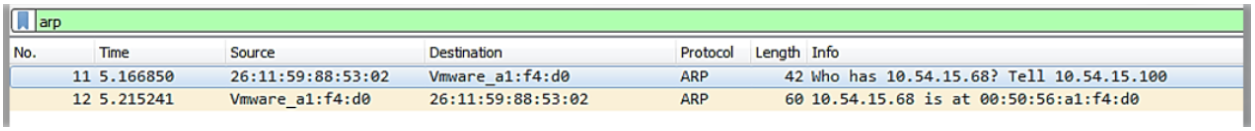
Normal ARP Traffic	Suspicious ARP Traffic
ARP broadcasts are normal from both clients and servers, including network devices at a reasonable flow.	Tens, hundreds, or even thousands of ARP broadcast messages in a small amount of time.
ARP Request typically follows a response, but it depends on who is making the request. A network device might send many ARP broadcasts into the network for various reasons.	Two identical MAC addresses in the network with different IP addresses.
Legitimate gratuitous ARP packets.	Gratuitous ARP packets sent by an attacker.

If we see 1 MAC address associated with 2 IPs or we see 1000s of ARP Replies in small amount of time, indicates ARP Poisoning or ARP spoofing, if we see 100s of ARP broadcast request that indicates someone is scanning for Active hosts.

So an attacker manipulate other hosts ARP cache Table by sending Gratuitous(large amount) ARP Replies which is ARP spoofing and it leads to MITM attack, so we tell the Victim that we are the Gateway and to the gateway that we are the victim so it sends its traffic through us.

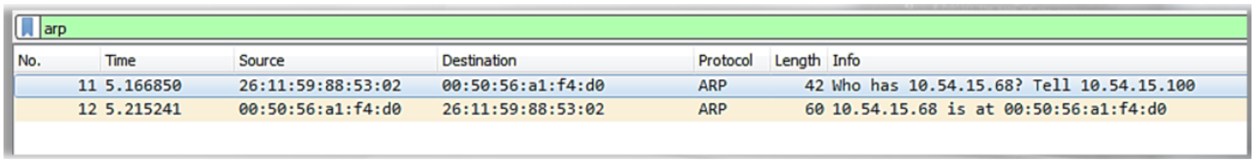
Normal ARP:

Here is SS of 2 packets: 1 ARP Request 1 ARP Response:



```
arp
No.        Time        Source              Destination         Protocol  Length  Info
        11 5.166850    26:11:59:88:53:02   Vmware_a1:f4:d0     ARP           42 Who has 10.54.15.68? Tell 10.54.15.100
        12 5.215241    Vmware_a1:f4:d0     26:11:59:88:53:02   ARP           60 10.54.15.68 is at 00:50:56:a1:f4:d0
```

To be able to see the MAC address for both Source and Destination, we have to make a quick change within wireshark: in **View > Name Resolution > Resolve Physical Addresses**

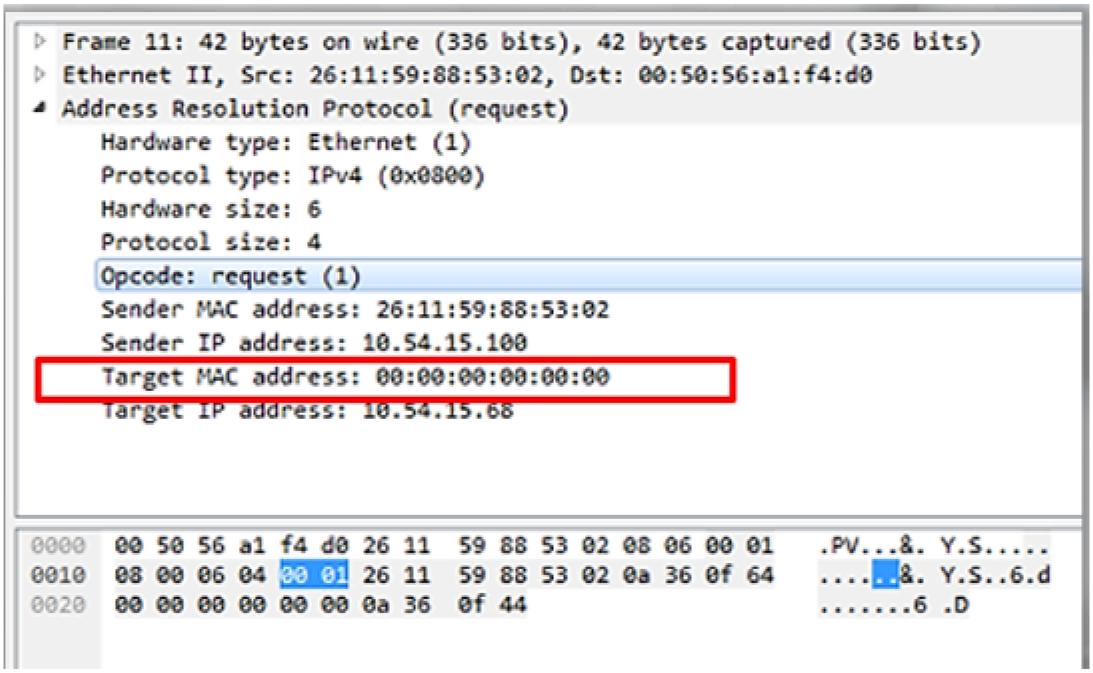


arp

No.	Time	Source	Destination	Protocol	Length	Info
11	5.166850	26:11:59:88:53:02	00:50:56:a1:f4:d0	ARP	42	Who has 10.54.15.68? Tell 10.54.15.100
12	5.215241	00:50:56:a1:f4:d0	26:11:59:88:53:02	ARP	60	10.54.15.68 is at 00:50:56:a1:f4:d0

Here we see the MAC for both Source and Destination now:

ARP request:

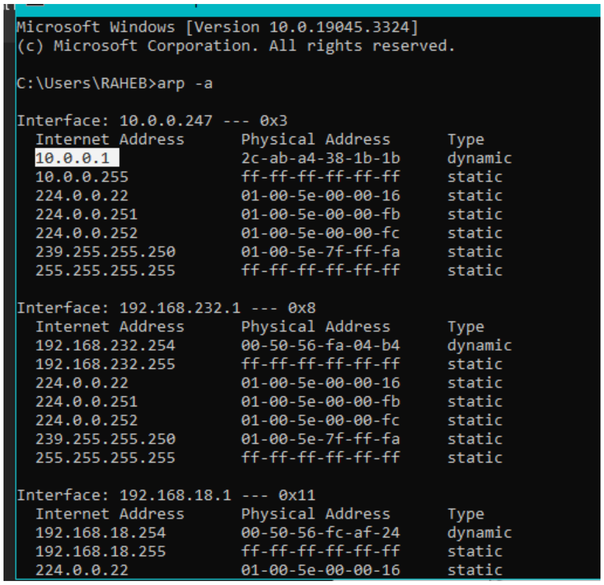


```
▷ Frame 11: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
▷ Ethernet II, Src: 26:11:59:88:53:02, Dst: 00:50:56:a1:f4:d0
◢ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 26:11:59:88:53:02
    Sender IP address: 10.54.15.100
    Target MAC address: 00:00:00:00:00:00
    Target IP address: 10.54.15.68

0000  00 50 56 a1 f4 d0 26 11  59 88 53 02 08 06 00 01   .PV...&. Y.S.....
0010  08 00 06 04 00 01 26 11  59 88 53 02 0a 36 0f 64   ......&. Y.S..6.d
0020  00 00 00 00 00 00 0a 36  0f 44                     .......6 .D
```

As we see the Opcode: request (1), this mean this is an ARP request. And we see the Sender IP which is 10.54.15.100, that mean this IP wants to find the MAC address of the IP 10.54.15.68. as we see the Target Mac is the broadcast Mac address 00:00:00:00:00:00, that mean its asking everyone saying who is IP 10.54.15.68.

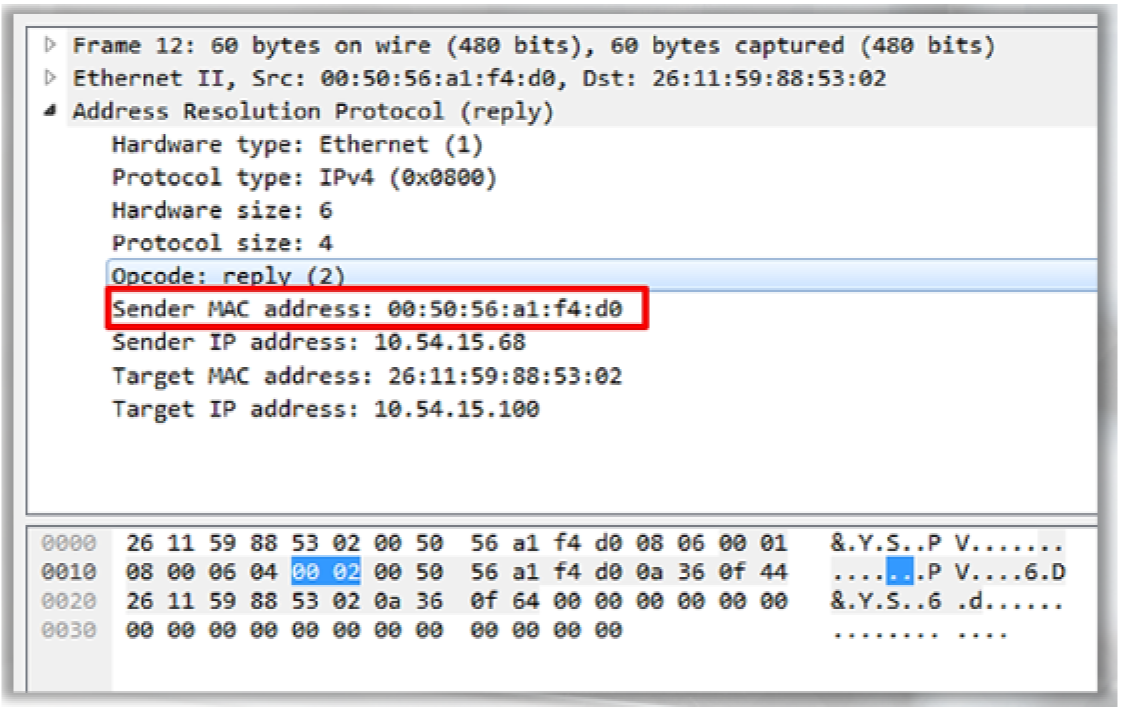
And if we want to see our mac table, then we can use the arp command like this:

Command: arp -a



```
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

C:\Users\RAHEB>arp -a

Interface: 10.0.0.247 --- 0x3
  Internet Address      Physical Address      Type
  10.0.0.1              2c-ab-a4-38-1b-1b     dynamic
  10.0.0.255            ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 192.168.232.1 --- 0x8
  Internet Address      Physical Address      Type
  192.168.232.254       00-50-56-fa-04-b4     dynamic
  192.168.232.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 192.168.18.1 --- 0x11
  Internet Address      Physical Address      Type
  192.168.18.254        00-50-56-fc-af-24     dynamic
  192.168.18.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
```

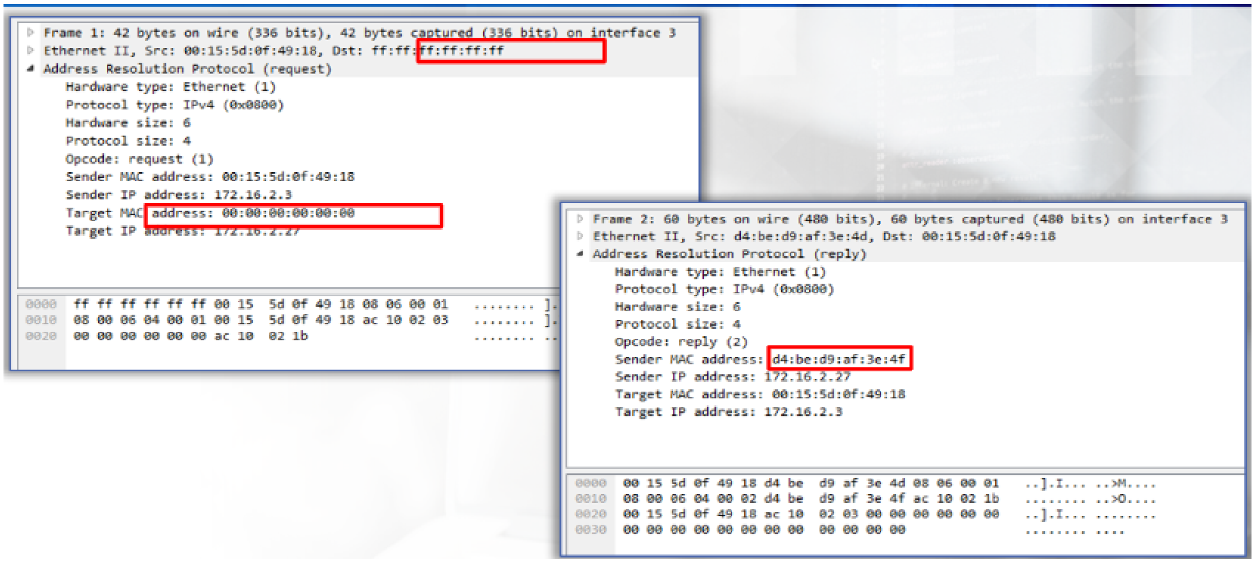
ARP Replay:



```
▷ Frame 12: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
▷ Ethernet II, Src: 00:50:56:a1:f4:d0, Dst: 26:11:59:88:53:02
◢ Address Resolution Protocol (reply)
     Hardware type: Ethernet (1)
     Protocol type: IPv4 (0x0800)
     Hardware size: 6
     Protocol size: 4
     Opcode: reply (2)
     Sender MAC address: 00:50:56:a1:f4:d0
     Sender IP address: 10.54.15.68
     Target MAC address: 26:11:59:88:53:02
     Target IP address: 10.54.15.100

0000  26 11 59 88 53 02 00 50  56 a1 f4 d0 08 06 00 01   &.Y.S..P V.......
0010  08 00 06 04 00 02 00 50  56 a1 f4 d0 0a 36 0f 44   .......P V....6.D
0020  26 11 59 88 53 02 0a 36  0f 64 00 00 00 00 00 00   &.Y.S..6 .d......
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

As we see the Opcode: reply (2), it indicates that it’s a ARP replay, which here we see it responded with its Mac Address as we see in the highlighted above. Now both hosts will add each other’s MAC address to their ARP table so next time they don’t do the ARP Request and Replay again and directly start the communication.

Here is another example:



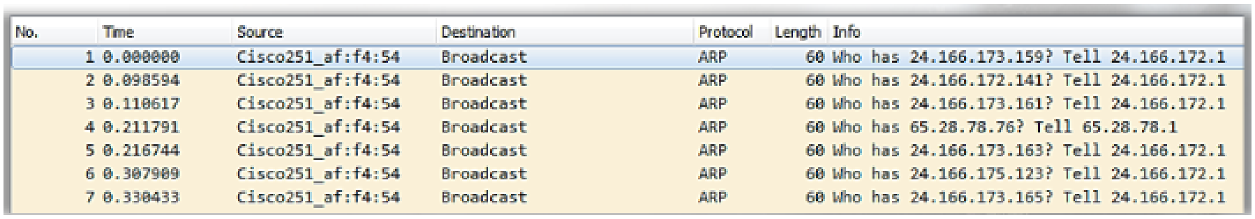
```
▷ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 3
▷ Ethernet II, Src: 00:15:5d:0f:49:18, Dst: ff:ff:ff:ff:ff:ff
◢ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 00:15:5d:0f:49:18
    Sender IP address: 172.16.2.3
    Target MAC address: 00:00:00:00:00:00
    Target IP address: 172.16.2.27

0000  ff ff ff ff ff ff 00 15  5d 0f 49 18 08 06 00 01   ........ ].
0010  08 00 06 04 00 01 00 15  5d 0f 49 18 ac 10 02 03   ........ ].
0020  00 00 00 00 00 00 ac 10  02 1b                     ........ ..
```

```
▷ Frame 2: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 3
▷ Ethernet II, Src: d4:be:d9:af:3e:4d, Dst: 00:15:5d:0f:49:18
◢ Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: d4:be:d9:af:3e:4f
    Sender IP address: 172.16.2.27
    Target MAC address: 00:15:5d:0f:49:18
    Target IP address: 172.16.2.3

0000  00 15 5d 0f 49 18 d4 be  d9 af 3e 4d 08 06 00 01   ..].I... ..>M....
0010  08 00 06 04 00 02 d4 be  d9 af 3e 4f ac 10 02 1b   ........ ..>O....
0020  00 15 5d 0f 49 18 ac 10  02 03 00 00 00 00 00 00   ..].I... ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

Suspicious ARP:

Normal: ARP broadcasts are normal from both clients and servers, and from network devices at reasonable flow.

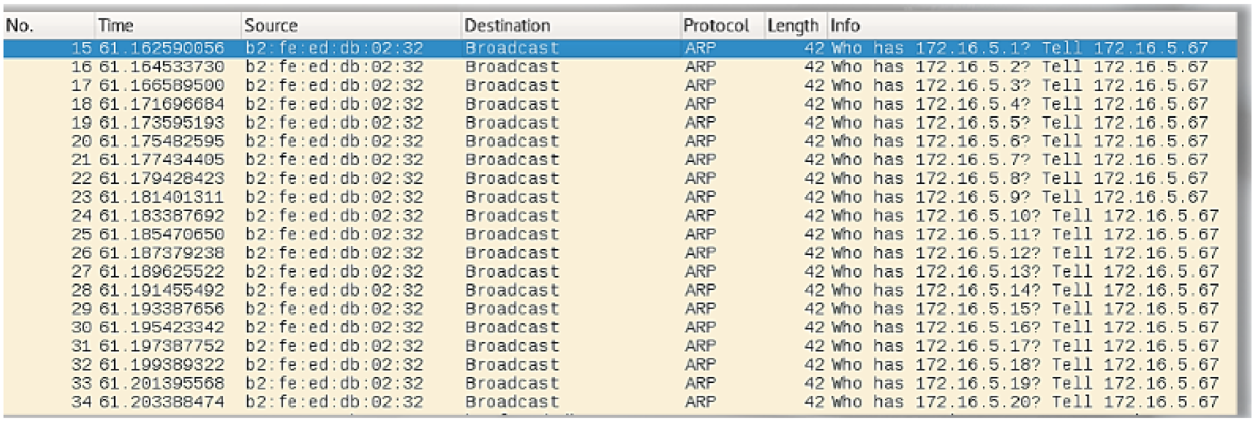
Suspicious: tens, hundreds or even thousands of ARP broadcast messages in a small amount of time.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 24.166.173.159? Tell 24.166.172.1
2	0.098594	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 24.166.172.141? Tell 24.166.172.1
3	0.110617	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 24.166.173.161? Tell 24.166.172.1
4	0.211791	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 65.28.78.76? Tell 65.28.78.1
5	0.216744	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 24.166.173.163? Tell 24.166.172.1
6	0.307909	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 24.166.175.123? Tell 24.166.172.1
7	0.330433	Cisco251_af:f4:54	Broadcast	ARP	60	Who has 24.166.173.165? Tell 24.166.172.1

Here we see 7 broadcast ARP requests , as the source looks like Cisco and its trying to find status of the network, mean its checking to see which IPs are active within the network.

How do we know if this is suspicious or not? Like is it misconfiguration on the cisco device? Or do we even have a Cisco device? Based on response to these questions, we might look into it further to confirm answers to these questions.



No.	Time	Source	Destination	Protocol	Length	Info
15	61.162590056	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.1? Tell 172.16.5.67
16	61.164533730	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.2? Tell 172.16.5.67
17	61.166589500	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.3? Tell 172.16.5.67
18	61.171696684	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.4? Tell 172.16.5.67
19	61.173595193	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.5? Tell 172.16.5.67
20	61.175482595	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.6? Tell 172.16.5.67
21	61.177434405	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.7? Tell 172.16.5.67
22	61.179428423	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.8? Tell 172.16.5.67
23	61.181401311	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.9? Tell 172.16.5.67
24	61.183387692	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.10? Tell 172.16.5.67
25	61.185470650	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.11? Tell 172.16.5.67
26	61.187379238	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.12? Tell 172.16.5.67
27	61.189625522	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.13? Tell 172.16.5.67
28	61.191455492	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.14? Tell 172.16.5.67
29	61.193387656	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.15? Tell 172.16.5.67
30	61.195423342	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.16? Tell 172.16.5.67
31	61.197387752	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.17? Tell 172.16.5.67
32	61.199389322	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.18? Tell 172.16.5.67
33	61.201395568	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.19? Tell 172.16.5.67
34	61.203388474	b2:fe:ed:db:02:32	Broadcast	ARP	42	Who has 172.16.5.20? Tell 172.16.5.67

This might look like scanning to see which IPs within the network is active and which are not, so those IPs that respond to those ARP requests are active within the network and those that wont respond that mean its inactive IP. We know indeed it’s a scan by looking at the IPs that increment by 1 and we see the time between each packet is small which indicate it’s a scan.

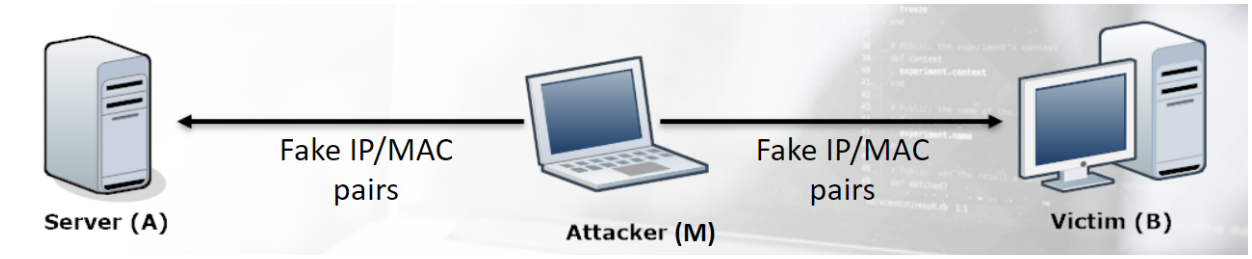
As these days its common to block ICMP within the network so the attacker wont be able to find which IPs are active within the network, and ARP is necessary within the network so we cant block it as if we do we wont be able make the devices within the network communicate with each other. So an attack can use ARP to discover hosts.

And attackers might change the mac address of the host they compromised within the network to make it look legit.

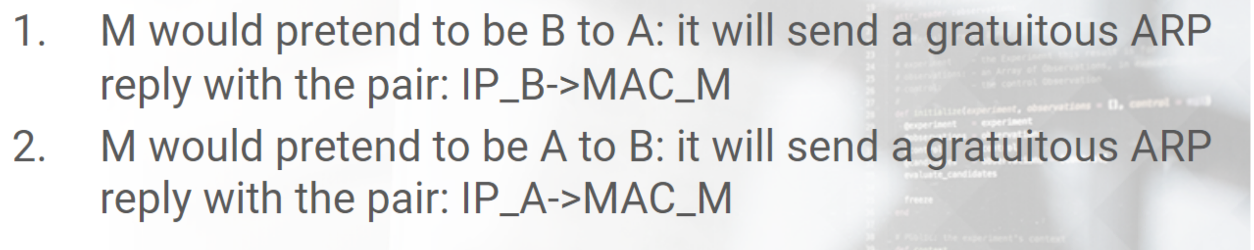
ARP Spoofing/cache Poisoning:

ARP poisoning, also known as ARP spoofing or ARP cache poisoning, is a type of cyberattack in which an attacker sends malicious Address Resolution Protocol (ARP) messages to associate their own MAC address with the IP address of a legitimate device on a local network.

If we see an ARP Replay that keeps happening over and over, its like a host keep telling another host their MAC address, that is ARP poisoning which leads to MITM attack. So its like telling the victim that we are the gateway so send your traffic to me.

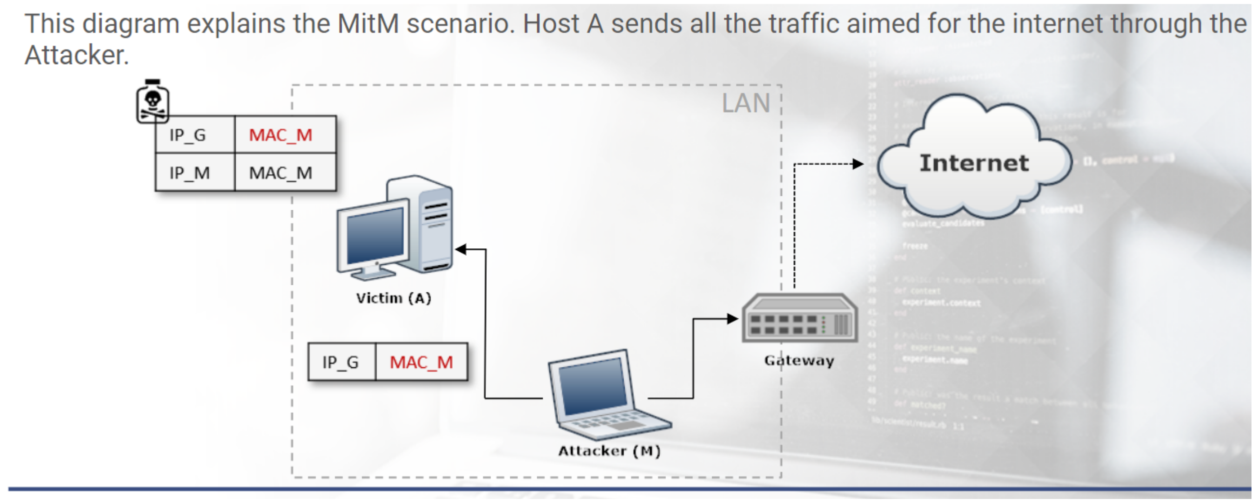


Here the attacker keep sending ARP response to both the A and B to poison the ARP cache, mean it keep telling the A that the attacker is the B, and telling B that the A is the attacker. It sends To A that the MAC of the B is the attacker MAC, mean it tells to the Server A that the IP of the victim B belong to the attacker MAC. And tell the B that the MAC address of the A is the attacker MAC, so it provides them with their own MAC for those hosts.



1. M would pretend to be B to A: it will send a gratuitous ARP reply with the pair: IP_B->MAC_M
2. M would pretend to be A to B: it will send a gratuitous ARP reply with the pair: IP_A->MAC_M

The attacker needs to keep sending those ARP Replies , as there is TTL in host ARP cache, its usually 30 sec. once the victim receives those ARP Responses, its cache will be poisoned, mean it will associate the MAC of the attacker with the IP of the legit host, it would be the IP of the Legit host and the MAC of the attacker cached in the ARP table of the victim. The attacker needs to this for the both , so it can intercept or look into both sides conversation.



This diagram explains the MitM scenario. Host A sends all the traffic aimed for the internet through the Attacker.

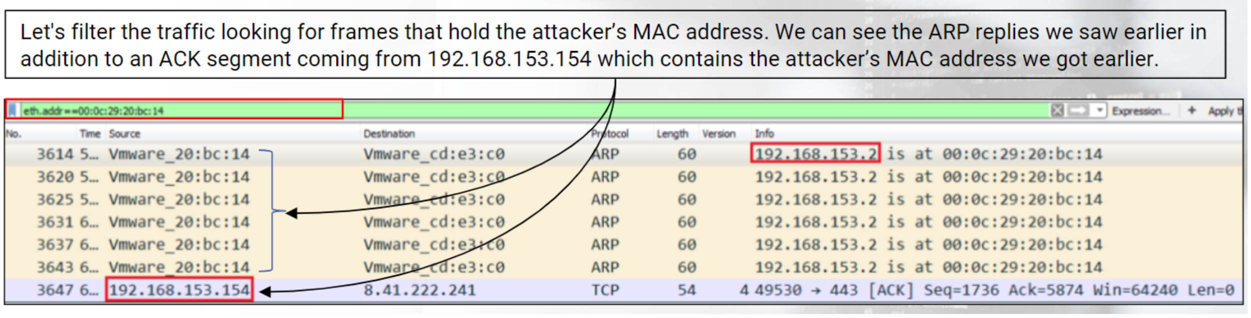
IP_G	MAC_M
IP_M	MAC_M

IP_G	MAC_M

Here we see a host is connected to the internet, and we know when a host trying to send traffic to an IP that dosnt exist in the network, it will send that traffic to the default gateway, mean as part of its Destination Mac address it will put the default Gateway MAC address and the IP of where this traffic is meant for, so now here the Attacker is telling the Victim A that the Mac address of the Gateway is the MAC of the attacker, so it sends its traffic through the attacker, and then the attacker can take it and forward it to the Gateway.

As we said that the ARP response should be keep happening as if not, the correct ARP response will be generated to restore the poisoned cached to correct values.

To detect this, we will look to see if one MAC is associated with 2 IPs. we can filter for that MAC and we will see all the IPs associated with that MAC:



Let's filter the traffic looking for frames that hold the attacker's MAC address. We can see the ARP replies we saw earlier in addition to an ACK segment coming from 192.168.153.154 which contains the attacker's MAC address we got earlier.

```
eth.addr==00:0c:29:20:bc:14
```

```
No.     Time Source             Destination        Protocol Length Version Info
   3614 5... Vmware_20:bc:14    Vmware_cd:e3:c0    ARP      60             192.168.153.2 is at 00:0c:29:20:bc:14
   3620 5... Vmware_20:bc:14    Vmware_cd:e3:c0    ARP      60             192.168.153.2 is at 00:0c:29:20:bc:14
   3625 5... Vmware_20:bc:14    Vmware_cd:e3:c0    ARP      60             192.168.153.2 is at 00:0c:29:20:bc:14
   3631 6... Vmware_20:bc:14    Vmware_cd:e3:c0    ARP      60             192.168.153.2 is at 00:0c:29:20:bc:14
   3637 6... Vmware_20:bc:14    Vmware_cd:e3:c0    ARP      60             192.168.153.2 is at 00:0c:29:20:bc:14
   3643 6... Vmware_20:bc:14    Vmware_cd:e3:c0    ARP      60             192.168.153.2 is at 00:0c:29:20:bc:14
   3647 6... 192.168.153.154    8.41.222.241       TCP      54       4     49530 → 443 [ACK] Seq=1736 Ack=5874 Win=64240 Len=0
```

Here we see that IP 192.168.153.154, has the same MAC as the *.2 IP, as we see there is a ARP response with a MAC that *.2 IP is associated with it, and the host 154 also have that MAC as well as we see in the filter, it shows it. So the 154 is trying to perform ARP poisoning to act as the *.2 host.