**CS6008 Cryptography and Network Security.**

**Assignment No.6**

**Name:** G Vignesh **Reg No:**2018103078

**Date:**5.6.2022

**Module-4:** NETWORK EXPLOITS

**External Learning-** Using arpspoof to poison network and detect using Wireshark

**Aim:**

Using Arpspoof tools , to perform poisoning of a victim systems-a arp table by changing the MAC address of communicating system-b with attackers system MAC address and capture the packets send from machine a to machine b. That is performing man in middle attack.Detecting this arpspoof attack using network analyser such as wireshark.

**Tools used:**

1.Virtual box

2.virtual linux systems

3. Ettercap(Arpspoof tool).

4.Wireshark packet analyzer

**Description:**

**ARP Protocol:**

ARP stands for **Address Resolution Protocol**, which helps a network host make a translation from the IP-address to the MAC-address. This is required in order for data to pass from the OSI model's [Network Layer](https://www.coengoedegebure.com/osi-model/#layer3) (layer 3) to the [Data Link layer](https://www.coengoedegebure.com/osi-model/#layer2) (layer 2) and vice-versa.

Suppose Machine A needs to transfer data to Machine B. Zooming in to the lower levels of the OSI model, it would need to pass through the Network layer, the Data Link layer and the [Physical layer](https://www.coengoedegebure.com/osi-model/#layer1) (layer 1). For Machine A to be able to address Machine B, Machine A would need to know the IP address of Machine B; information that is known in the Network layer.

The Data Link layer communicates using MAC addresses. So, a conversion needs to take place from the IP address to the MAC address of Machine B (and vice-versa on the recipient machine).

The conversion from, or rather resolution of, the IP address into MAC address (and the other way around) is where the ARP protocol comes into play. Both machines will have an **ARP table** where the IP- and corresponding MAC-addresses of all known machines are stored. Then how does Machine A get the MAC-address corresponding to the IP Address of Machine B?

Machine A will just ask for it.

**Step-1: Victim Network analysis**

**ARP table Updation:**

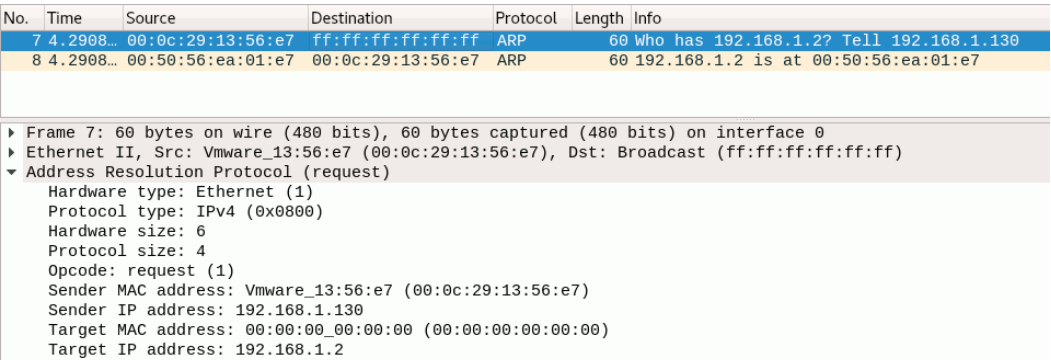
Initially this ARP-table is empty.

Let's briefly go over the 3 steps:

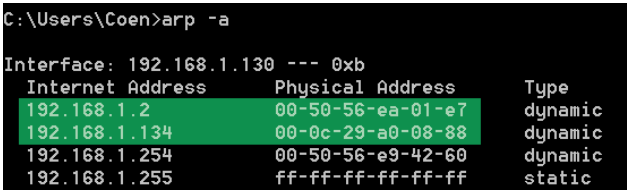
1. In the first step of the ARP protocol, Machine A sends out an **ARP request**. This is a broadcast to the network with the question "Who has the MAC-address for the IP-address of Machine B?".
2. The machine that has this knowledge (usually Machine B itself), will send an **ARP reponse** stating "MAC-address B is the MAC-address of Machine B".
3. Machine A receives the ARP response and writes (or updates) the entry in his **ARP table**.
4. The last step is exactly where the problem with this protocol lies. It accepts ARP response from whoever says it’s the IP address is this and corresponding MAC is this without cross checking or verifying.

**ARP Request:** Wireshark shows ip address 192.168.1.130 asks who has ip address 192.168.1.2

**ARP Response:** Ip address 192.168.1.2 responds 192.168.1.2 is at MAC address of 00:50:56:ea:01:e7

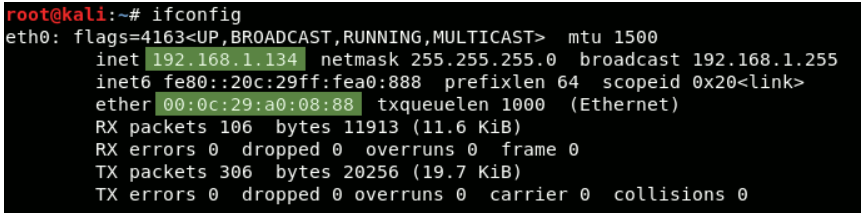


Now when we look at ARP table of 192.68.1.130 it gets updated.



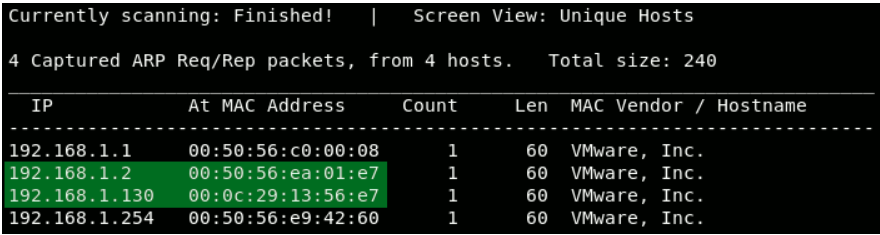
**Step-2: Setting Up Attacker**

**192.168.1.2 Machine –A 192.168.1.130 Machine-B 192.168.1.134-Attacker**



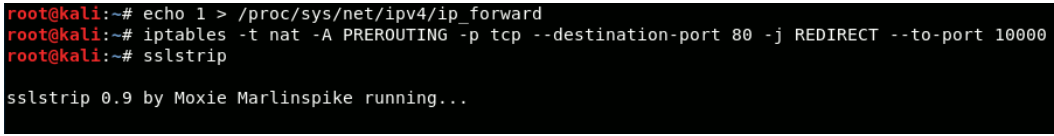
We are setting up a attacker network 192.168.1.134 whose MAC is 00:0c:29:08:88 within the same network.

Victim networks Machine-A and Machine-b are in the same network which can be viewed by **netdiscover -r 192.168.1.0/24 command**

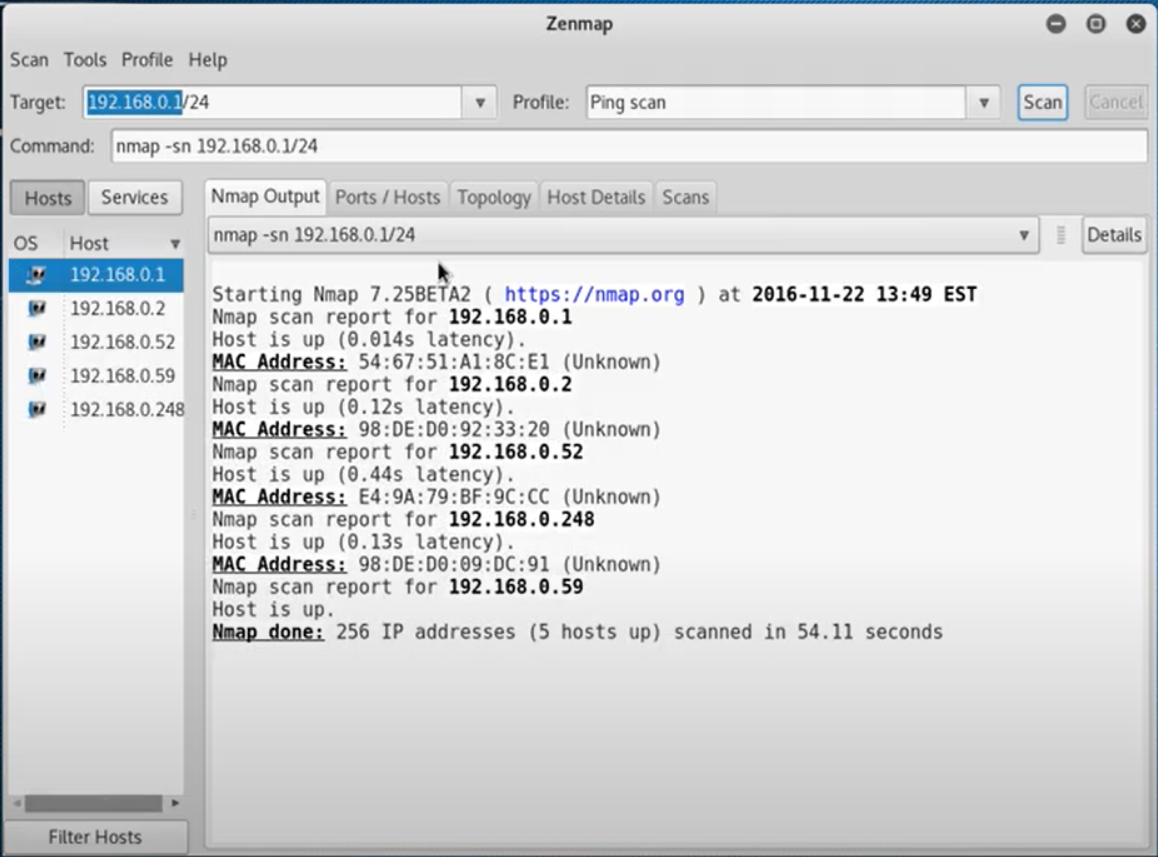


Now we know the IP-addresses of the machines we want to be in between of, we are almost ready to perform the ARP-poisoning. Before we do that however, there are a few things we need to do to setup SSLStrip in preparation of the attack.  
 Make sure all IPv4 traffic is forwarded. If we wouldn't do this, all IPv4 traffic would stop at our hacker machine in the middle of both nodes. This would result in a Denial-of-Service attack to our victim, who would no longer able to communicate to the Gateway.

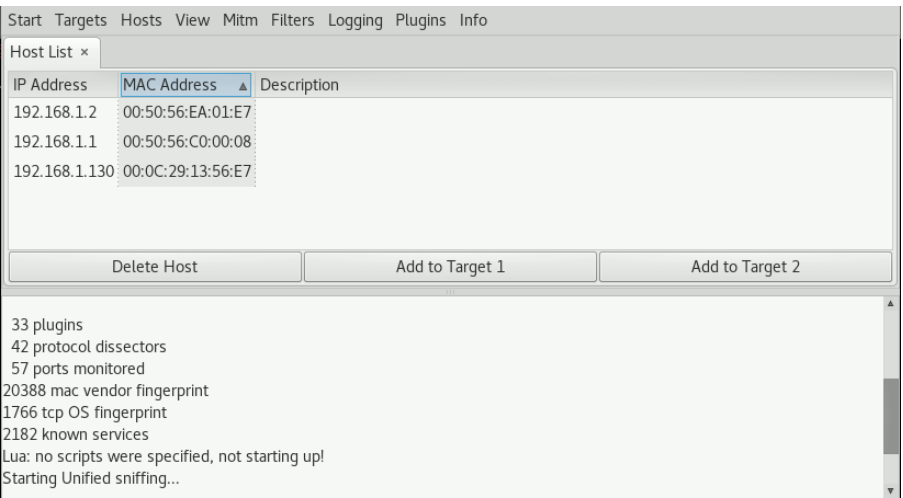
We also need to make sure all HTTP-traffic is redirected to SSLStrip. When running, SSLStrip listens on port 10000 by default. Redirecting incoming HTTP-traffic (TCP on port 80) to port 10000 on our own machine, requires a modification in the linux firewall tables.   
Since sslstrip is included in Kali Linux, running this program is as easy as executing the command sslstrip.



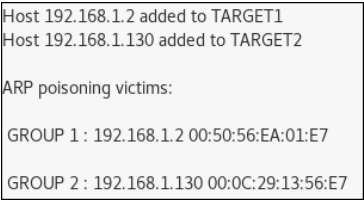
**Step-3:** **Performing ARP Poisoning.**



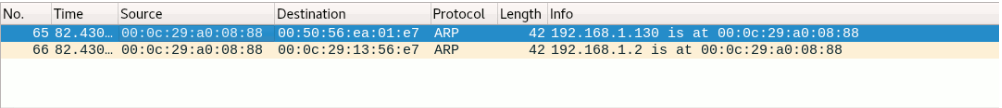
Ettercap is the tool used for ARP poisoning , without setting target-1 and target -2 that is the victims between which man in middle attack has to take place it generally broadcasts all ip address with its MAC address and fool all the machines in the network.



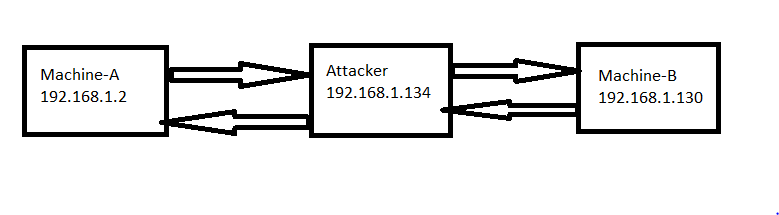
It shows all the Machines in our network , Machine A is set as target 1 and Machine b as target2



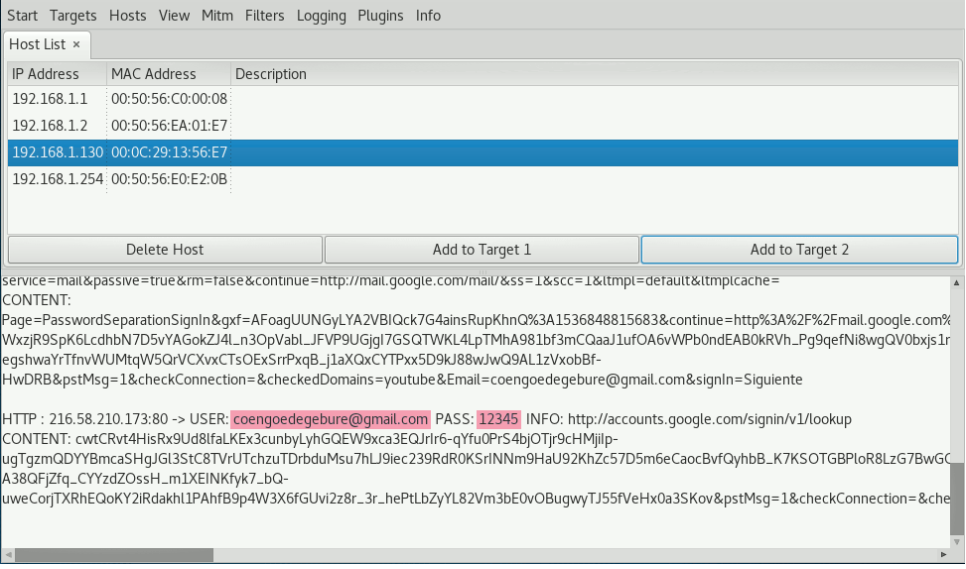
Wireshark shows how this ARP Request and response happened.



Attcaker is responding to both Machine a and Machine B with its own MAC address.



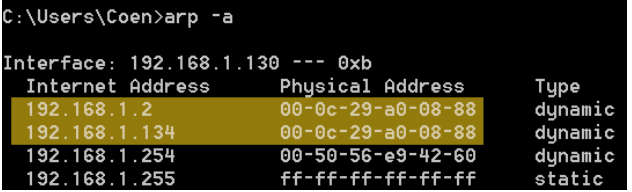
Now the attacker is in middle of machine a and machine b.



Any HTTP unsecure request between Machine a and Machine b can be viewed by attacker in its Ettercap tool.

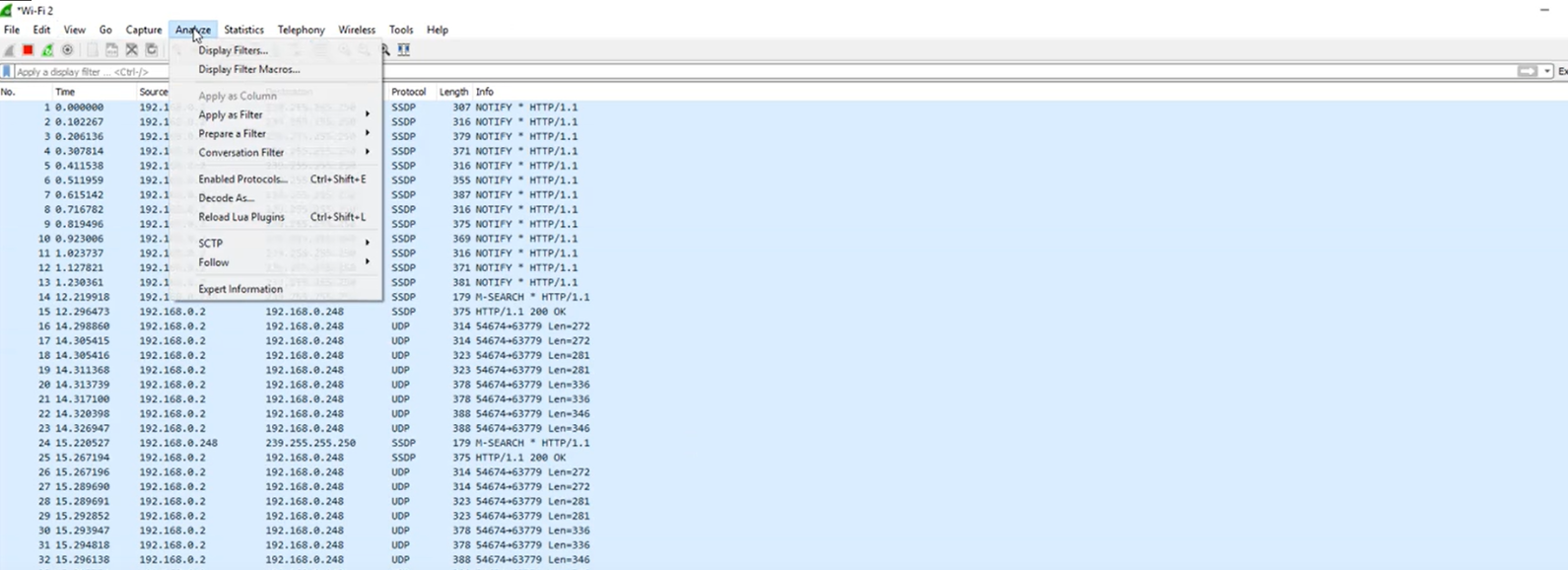
**Step-4:**  **Detecting Arp Poisoning**

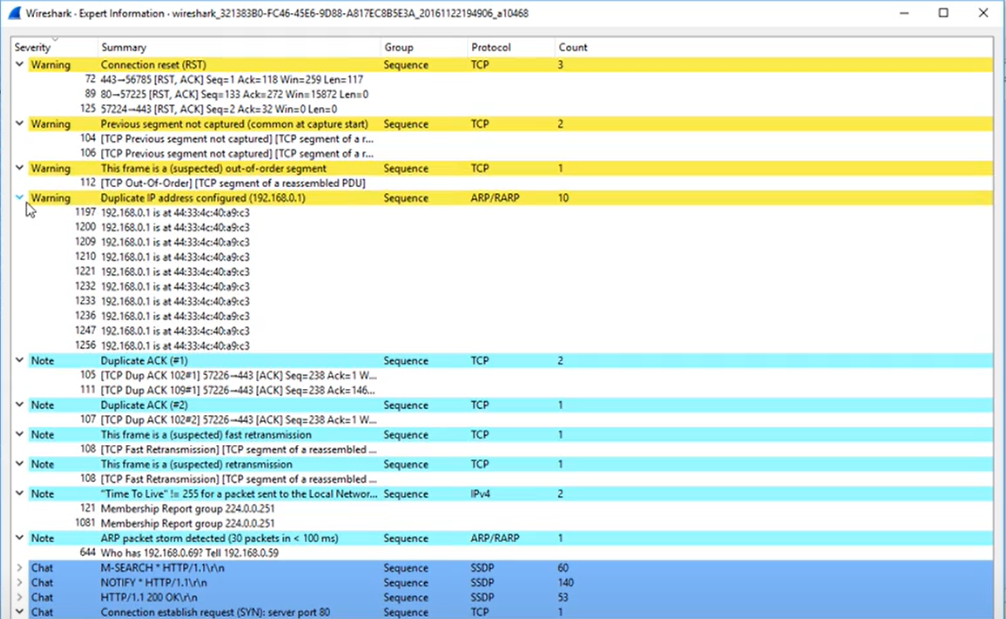
Victims can identify the ARPSpoof attacks by looking into their arp table and if there are any duplicate address we can conclude that some attacker has poisoned the arp table with his own MAC address.

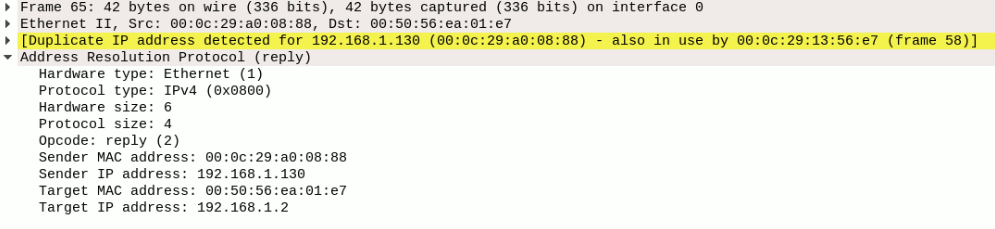
****

**WireShark** Can also be used to easily identify these attacks. Navigate to

**ANALYZE 🡪Expert Information**







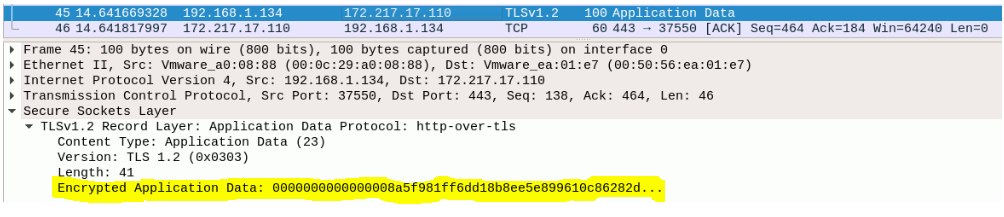
Wireshark shows warning for Duplicate address identified in the network. By this way of analyzing the network with wireshark we can easily identify any ARPspoof attacks happened or not in our network.

**Conclusion:**

ARP Spoofing is one way to perform a man-in-the-middle attack. However, there are a few security countermeasures to protect you against this.

1. ARP is a protocol that relies on IPv4. In IPv6 ARP has been replaced by the **Neighbor Discovery Protocol** ([NDP](https://en.wikipedia.org/wiki/Neighbor_Discovery_Protocol)), however the [adoption rate of IPv6](https://www.google.com/intl/en/ipv6/statistics.html) is still relatively low. The security extension of this protocol (Secure Neighbor Discovery or [SEND](https://en.wikipedia.org/wiki/Secure_Neighbor_Discovery)) uses cryptography to ensure that the claimed source of an NDP message is the owner of the claimed address.
2. Most modern **switches** come with an ARP spoofing protection feature that can be enabled to prevent this attack.
3. Creating a **static ARP entry** in your server can help reduce the risk of spoofing. If you have two hosts that regularly communicate with one another, setting up a static ARP entry creates a permanent entry in your ARP cache that can help add a layer of protection from spoofing.

The better option is to make our website Secured using HTTPS, so that even if ARPspoof attack happens the attack can see only the encrypted data and do nothing with it.



ARPspoof attack which can take place only within a network and by allowing only authorized people into our network say office network or wifi network in our houses this attack cannot take place, so now a days these attacks or happening and not a big concern in the industry now a days.