Sniffing

Objectives:

- Become aware of a class of vulnerabilities known as sniffing.
- Learn how to use a sniffer tool.

Definition of Sniffing:

- A program or device that captures vital information from the network traffic specific to a particular network.
- Sniffing is a data interception technology
- The objective of sniffing is to steal:
  - Passwords (from email, the web, SMB, ftp, SQL, or telnet).
  - Email text.
  - Files in transfer (email files, ftp files, or SMB).
- There are other goals for sniffing like network maintenance.

So, a **packet sniffer** is: a piece of software that grabs all of the traffic flowing into and out of a computer attached to a network.

Protocols Vulnerable to Sniffing:

Protocols that are susceptible to sniffers include:

- Telnet and Rlogin: Keystrokes including user names and passwords.
- HTTP: Data sent in clear text.
- SMTP: Passwords and data sent in clear text.
- NNTP: Passwords and data sent in clear text.
- POP: Passwords and data sent in clear text.
- FTP: Passwords and data sent in clear text.
- IMAP: Passwords and data sent in clear text.

Types of Sniffing:

There are two types of sniffing:

Passive sniffing: Sniffing through a Hub.

Active sniffing: Sniffing through a Switch.

Passive Sniffing:

- It is called passive because it is difficult to detect.
- "Passive sniffing" means sniffing through a hub.
- Attacker simply connects the laptop to the hub and starts sniffing.

Active Sniffing:

- Sniffing through a switch.
- Difficult to sniff.
- Can easily be detected.
- Techniques for active sniffing:
  - ARP (Address Resolution protocol) spoofing.
  - MAC flooding.

What is Address Resolution protocol?

- Address Resolution Protocol is a network layer protocol used to convert an IP address to a physical address (called a MAC address), such as an Ethernet address.
- To obtain a physical address, the host broadcasts an ARP request to the TCP/IP network
- The host with the IP address, in the request, replies with its physical hardware address on the network.

ARP Spoofing Attack:

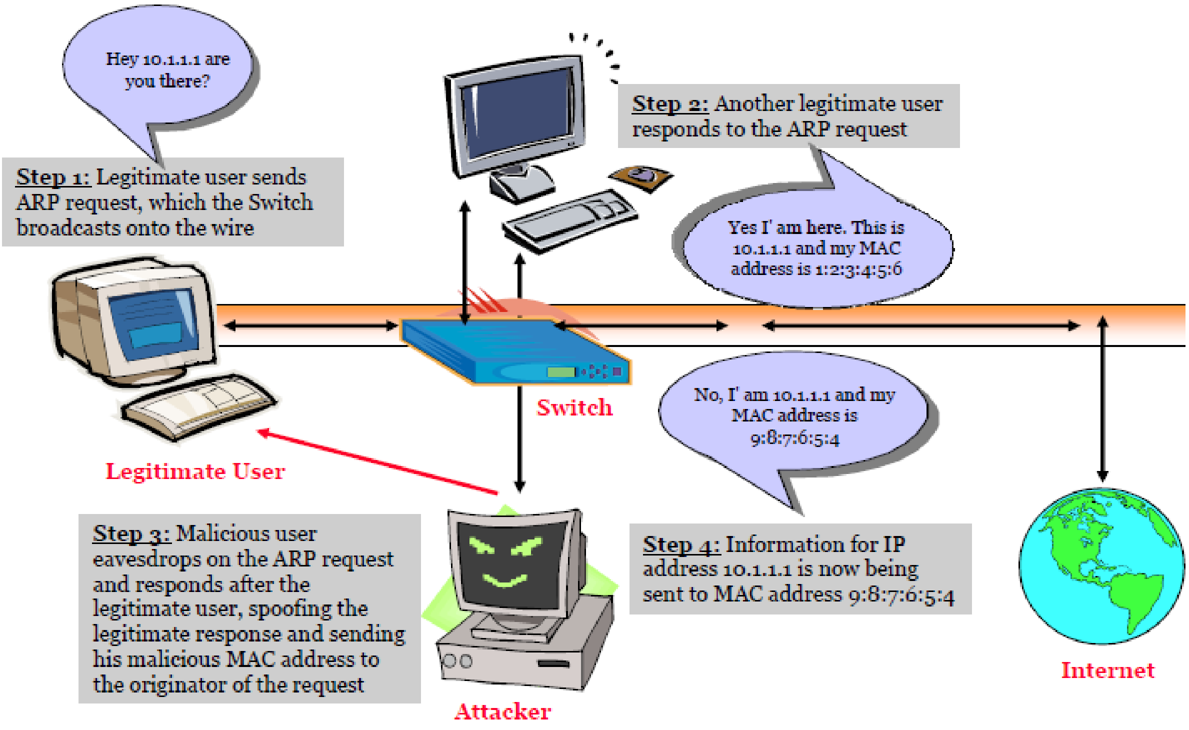
- ARP resolves IP addresses to the MAC (hardware) address of the interface to send data.

- ARP packets can be forged to send data to the attackers' machines.
- An attacker can exploit ARP poisoning to intercept network traffic between two machines on the network.
- By MAC flooding a switch's ARP table with spoofed ARP replies, the attacker can overload the switches and then packet sniff the network while the switch is in "forwarding mode".
- The most common tool for ARP Spoofing used in Linux and windows is **Ettertap**.

How Does ARP Spoofing (Poisoning) Work?

When a legitimate user initiates a session with another user in the same Layer 2 broadcast domain, an address resolution protocol (ARP) request is broadcasted using the recipient's IP address and the sender waits for the recipient to respond with a MAC address

A malicious user eavesdropping on this unprotected Layer 2 broadcast domain can respond to the broadcast ARP request and reply to the sender by spoofing the intended recipient's MAC address. See the figure below.


Hey 10.1.1.1 are you there?
Yes I' am here. This is 10.1.1.1 and my MAC address is 1:2:3:4:5:6
No, I' am 10.1.1.1 and my MAC address is 9:8:7:6:5:4
Legitimate User
Switch
Attacker
Internet

Step 1: Legitimate user sends ARP request, which the Switch broadcasts onto the wire

Step 2: Another legitimate user responds to the ARP request

Step 3: Malicious user eavesdrops on the ARP request and responds after the legitimate user, spoofing the legitimate response and sending his malicious MAC address to the originator of the request

Step 4: Information for IP address 10.1.1.1 is now being sent to MAC address 9:8:7:6:5:4

Threats of ARP Poisoning:

- Internal network attacks are typically operated via ARP Poisoning attacks
- Everyone can download on the Internet Malicious software used to run ARP Spoofing attacks
- Using fake ARP messages, an attacker can divert all communication between two machines so that all traffic is exchanged via his PC
- By means, such as a man-in-the-middle attack, the attacker can, in particular:
  - Run Denial of Service (DoS) attacks.
  - Intercept data.
  - Collect passwords.
  - Manipulate data.
  - Tap VoIP phone calls.

MAC Flooding:

- MAC flooding involves flooding the switch with numerous requests.
- Switches have a limited memory for mapping various MAC addresses to the physical ports on the switch.
- MAC flooding makes use of this limitation to bombard the switch with fake MAC addresses until the switch cannot keep up.
- The switch then acts as a hub by broadcasting packets to all the machines on the network.
- After this, sniffing can be easily performed.

The most common tool for MAC Flooding used in Linux and windows is **Etherflood.**

DHCP Starvation Attack:

- A DHCP starvation attack works by broadcasting DHCP requests with spoofed MAC addresses
- This is easily achieved with attack tools such as gobbler

- If enough requests are sent, the network attacker can exhaust the address space available to the DHCP servers for a period of time
- The network attacker can then set up a rogue DHCP server on his or her system and respond to new DHCP requests from clients on the network.
- By placing a rogue DHCP server on the network, a network attacker can provide clients with addresses and other network information
- Since DHCP responses typically include default gateway and DNS server information, the network attacker can supply his or her own system as the default gateway and DNS server resulting in a "man-in-the-middle" attack.

DNS Poisoning Techniques:

- The substitution of a false Internet provider address at the domain name service level (e.g., where web addresses are converted into numeric Internet provider addresses)
- DNS poisoning is a technique that tricks a DNS server into believing it has received authentic information when, in reality, it has not
- Types of DNS Poisoning:
  1. Intranet DNS Spoofing (Local network).
  2. Internet DNS Spoofing (Remote network).
  3. Proxy Server DNS Poisoning.
  4. DNS Cache Poisoning.

1. Intranet DNS Spoofing (Local Network)

For this technique, you must be connected to the local area network (LAN) and be able to sniff packets. It works well against switches with ARP poisoning the router.

2. Internet DNS Spoofing (Remote Network)

Send a Trojan to the victim’s machine and change her DNS IP address to that of the attacker’s. It works across networks.

3. Proxy Server DNS Poisoning

Send a Trojan to the victim's machine and change her proxy server settings in Internet Explorer to that of the attacker's. it works across networks.

4. DNS Cache Poisoning

Normally, a networked computer uses a DNS server provided by the computer user's organization or an Internet service provider (ISP). DNS servers are generally deployed in an organization's network to improve resolution response performance by caching previously obtained query results. Poisoning attacks on a single DNS server can affect the users serviced directly by the compromised server or indirectly by its downstream server(s) if applicable.

To perform a cache poisoning attack, the attacker exploits a flaw in the DNS software. If the server does not correctly validate DNS responses to ensure that they are from an authoritative source (for example by using DNSSEC) the server will end up caching the incorrect entries locally and serve them to other users that make the same request.

This technique can be used to direct users of a website to another site of the attacker's choosing. For example, an attacker spoofs the IP address DNS entries for a target website on a given DNS server, replacing them with the IP address of a server he controls. He then creates files on the server he controls with names matching those on the target server. These files could contain malicious content, such as a computer worm or a computer virus. A user whose computer has referenced the poisoned DNS server would be tricked into accepting content coming from a non-authentic server and unknowingly download malicious content.

How to Detect Sniffing?

- You will need to check which machines are running in promiscuous mode.
- Run ARPWATCH and notice if the MAC address of certain machines has changed (Example: router's MAC address).
- Run network tools like HP OpenView and IBM Tivoli network health check tools to monitor the network for strange packets.

Countermeasures:

- Restriction of physical access to network media ensures that a packet sniffer cannot be installed.
- The best way to be secured against sniffing is to use Encryption. It would not prevent a sniffer from functioning but will ensure that what a sniffer reads is not important.
- ARP Spoofing is used to sniff a switched network, so an attacker will try to ARP spoof the gateway. This can be prevented by permanently adding the MAC address of the gateway to the ARP cache.
- Another way to prevent the network from being sniffed is to change the network to SSH.
- There are various methods to detect a sniffer in a network:
    - **Ping method:** This method relies on a problem in the target machine's kernel. We can construct an ICMP echo request with the IP address of the machine suspected of hosting a sniffer but with a deliberately mismatched MAC address. We send an ICMP echo packet to the target with the correct destination IP address, but a bogus destination hardware address. Most systems will disregard this packet since its hardware address information is incorrect. But in some Linux, NetBSD and NT systems, since the NIC is in promiscuous mode, the sniffer will grab this packet off the network as a legitimate packet and respond accordingly. If the target in question replies to our request, we know it is in promiscuous mode. Clever attackers are of course aware of this and can update their sniffers to filter out such packets as the NIC itself would have had it not been in promiscuous mode.
    - **ARP method:** We send out an ARP request to our target with all valid information except a bogus destination hardware address. A machine that is not in promiscuous mode would never see the packet, since it wasn't destined to them, therefore it wouldn't reply. If a machine is in promiscuous

mode, the ARP request would be seen and the kernel would process it and reply. By the machine replying, we know it is in promiscuous mode.

- **Latency method:** In this method, we ping the target and note the round trip time (RTT), from there. We create hundreds of fake TCP connections on our network segment at a lightning rate. We expect the sniffer to be processing those packets at a rate where the target machine's network latency will increase. We then ping the target once again, and compare the RTT this time to the first time. After a series of tests and averages, we can conclude whether or not a sniffer is indeed running on the target.
- **Using IDS (Intrusion detection system).**

- There are various tools to detect a sniffer in a network:
  - **Antisniff:** is tool can detect machines on the network that are running in promiscuous mode.
  - **ArpWatch:** is a tool that monitors Ethernet activity and keeps a database of Ethernet/IP address pairings.

- **In Small Network:**

  Use of static IP addresses and static ARP tables which prevents hackers from adding spoofed ARP entries for machines in the network.

- **In Large Networks:**
  - Network switch Port Security features should be enabled.
  - Use of ArpWatch to monitor Ethernet activity.

Lab Experiment

Requirements:

Setup a network contains at least two machines (in the lab) or you can use software like VMware or Virtual PC to build you virtual lab (in the home).

Procedures :

1. From PC1 setup Wireshark (or other sniffing tool), from **capture** menu select **interfaces** then a list of network interfaces NIC will show , select the interface that connected to the network you want to sniff.
2. Click **start** to start sniffing , in this case sniffing tool will capture any packet on the wire and analyze it ; as shown in figure 1 and 2 .
3. If we want to sniff a specific machine then we can use options of the tool to determine specific IP address to filter the captured packet to that IP address.

NOTE: We can use Wireshark sniffer using Backtrack Linux, From **Start→Backtrack→Privilege Escalation → Sniffers→Wireshark**.

Exercise:

1. Install Wireshark on your PC and start sniffing for some seconds, then analyze **four** different packets, for example if you capture ARP request packet you must show that this packet work only in the lower three layers and show the source IP and MAC and destination IP and MAC , also anything important and can recognize the captured packet.
2. Repeat step 1 using another sniffing tool like **Ettercap**, **dsniff** or any tool you may find it in the internet.
3. Just in one paper, talk about Ettercap or Etherflood tool.

***