# Module 08: Sniffing - Practical Guide

This guide covers the specialized tools and hands-on laboratory sessions required to master network sniffing and Man-in-the-Middle (MITM) attacks for the CEH v13 certification.

### 1. Comprehensive Sniffing Toolset

### Packet Analyzers & General Sniffers

* **Wireshark:** The gold standard. Used for deep packet inspection and analyzing captured traffic.
* **Tcpdump:** A powerful command-line packet analyzer for Linux.
* **TShark:** The terminal-based version of Wireshark.

### Man-in-the-Middle (MITM) & ARP Poisoning

* **Ettercap:** A comprehensive suite for MITM attacks, featuring a GUI and automated "Unified Sniffing" modes.
* **Bettercap:** The modern, modular replacement for Ettercap. It is more stable and powerful for WiFi, Bluetooth, and Ethernet attacks.
* **Arpspoof (part of dsniff):** A simple, reliable command-line tool specifically for redirecting traffic via ARP.

### Layer 2 & Protocol Attack Tools

* **Yersinia:** Specifically designed to exploit weaknesses in protocols like DHCP, STP, VTP, and HSRP.
* **macof:** Used for MAC Flooding attacks to overflow a switch's CAM table.
* **DHCPExplorer:** Used to discover rogue DHCP servers on a network.

## 2. Hands-On Lab Sessions

### Lab 1: Capture Cleartext Credentials (Wireshark)

**Goal:** Extract a password from an unencrypted HTTP/Telnet session.

1. **Launch Wireshark** and select your interface (eth0 or wlan0).
2. **Filter for traffic:** In the filter bar, type http or telnet.
3. **Perform Activity:** On the target machine (or your own), log into a site using http://.
4. **Analyze:** Find a POST request in the packet list.
5. **Follow Stream:** Right-click the packet -> **Follow -> TCP Stream**. The username and password will be visible in the text window.

### Lab 2: ARP Poisoning / MITM (Ettercap)

**Goal:** Intercept traffic between a victim and the gateway.

1. **Open Ettercap:** Run sudo ettercap -G for the GUI.
2. **Unified Sniffing:** Select **Sniff -> Unified Sniffing** and choose your interface.
3. **Host Discovery:** Go to **Hosts** -> **Scan for hosts**. Then **Hosts -> Host List**.
4. **Target Selection:** \* Select the Gateway IP and click **Add** to **Target 2**.
   * Select the Victim IP and click **Add to Target 1**.
5. **Attack:** Select **MITM** -> **ARP Poisoning**. Check "Sniff remote connections."
6. **Start:** Select **Start -> Start sniffing**.
7. **Verification:** All traffic from the victim to the internet now flows through your Parrot OS machine.

### Lab 3: MAC Flooding (macof)

**Goal:** Force a switch into "Fail-Open" mode to turn it into a hub.

1. **Identify Interface:** Ensure you are connected to a physical switch lab environment.
2. **Run Flood:**  
   sudo macof -i eth0 -n 100000
3. **Observation:** Use wireshark on a separate machine connected to the same switch. If successful, you will start seeing traffic destined for *other* MAC addresses appearing on your screen.

### Lab 4: DHCP Starvation Attack (Yersinia)

**Goal:** Exhaust the DHCP pool to deny service to new users.

1. **Launch Yersinia:** Run sudo yersinia -g for the GUI.
2. **Select Protocol:** Click the **DHCP** tab.
3. **Launch Attack:** Click **Launch Attack** and select **Sending RAW packets** or **Discover packets**.
4. **Monitor:** Watch the "Packets Sent" count rise.
5. **Verify:** Try to connect a new device to the network; it will fail to receive an IP address because the "Starvation" has consumed the entire pool.

### Lab 5: DNS Spoofing (Bettercap)

**Goal:** Redirect a victim from google.com to your malicious IP.

1. **Start Bettercap:** sudo bettercap -iface eth0.
2. **Set Targets:** set arp.spoof.targets [Victim\_IP].
3. Configure DNS: ```bash  
   set dns.spoof.domains https://www.google.com/search?q=google.com  
   set dns.spoof.address