WI-FI Hacking Notes

Wireless Adapter Connection to Kali Linux

When performing Wi-Fi penetration testing with Kali Linux, it's essential to use a wireless adapter that supports monitor mode and packet injection. Below is a guide on connecting a wireless adapter to Kali and ensuring it's ready for penetration testing.

1. Connecting the Wireless Adapter to Kali Linux

Steps:

1. **Plug in the Wireless Adapter**: Insert the wireless adapter into a USB port on your computer or laptop running Kali Linux.
2. **Check if Kali Recognizes the Adapter:**

- Open a terminal and type the command:
- **lsusb**
- This command lists all USB devices connected to your system. Look for an entry related to your wireless adapter (e.g., "Realtek Semiconductor Corp." or "Atheros Communications, Inc.").

1. **Ensure the Adapter is Enabled:**

- To check if your wireless adapter is recognized and enabled, use:
- **ifconfig**
- Look for an interface named something like wlan0, wlan1, etc. This represents your wireless interface.

1. **Installing Drivers (If Needed):**

- Most wireless adapters work out-of-the-box with Kali Linux, but in some cases, you might need to install drivers. If your adapter is not recognized, you may need to install the appropriate drivers for your wireless chipset.
- For example, for Realtek chipsets:
- **sudo apt-get update**
- **sudo apt-get install realtek-rtl88xxau-dkms**

2. Testing Monitor Mode Capabilities

Monitor mode allows your wireless adapter to capture all packets on a network, even those not addressed to your device. Here's how to enable and test it:

1. **Enable Monitor Mode:**

- Use **airmon-ng** to enable monitor mode on your wireless adapter.
- **sudo airmon-ng start wlan0**
- This command will change the mode of the interface from wlan0 to something like wlan0mon (monitor mode).

1. **Check Monitor Mode:**

- To verify that your adapter is in monitor mode, use:
- **iwconfig**
- Look for your interface (wlan0mon). It should now be listed with Mode: Monitor.

1. **Start Capturing Packets:**

- Test the monitor mode by capturing packets with **airodump-ng**.
- **sudo airodump-ng wlan0mon**
- This command should start showing you all the wireless networks in the vicinity.

Understanding MAC Address

1. What is a MAC Address?

- MAC (Media Access Control) Address: A unique identifier assigned to network interfaces for communications at the data link layer of a network segment.
- Format: Typically displayed as six groups of two hexadecimal digits, separated by colons (e.g., 00:1A:2B:3C:4D:5E).

Key Points:

- Permanence: MAC addresses are usually hardcoded into a device's network interface card (NIC) by the manufacturer.
- Uniqueness: Each device's MAC address should be unique, though some devices allow MAC address spoofing.

2. Finding Your MAC Address

Commands:

- To find the MAC address of your wireless adapter:
- **ifconfig wlan0** | grep ether
- This command will output something like:
- mathematical
- **ether 00:1A:2B:3C:4D:5E txqueuelen 1000 (Ethernet)**

3. MAC Address Spoofing

- Spoofing: Changing the MAC address of your network interface to disguise your device or bypass certain network filters.
- Command:
- **sudo ifconfig wlan0 down**
- **sudo ifconfig wlan0 hw ether 00:11:22:33:44:55**
- **sudo ifconfig wlan0 up**
- The above commands temporarily change your MAC address. Replace wlan0 with your interface name and 00:11:22:33:44:55 with the desired MAC address.

Wireless Modes: Managed & Monitor

1. Managed Mode

- Definition: The default mode for most wireless adapters, where the adapter connects to an access point (AP) and communicates with the network like a typical client.
- Use Case: Used for everyday Wi-Fi activities such as browsing the internet, downloading files, or streaming.

Commands:

- Checking Mode:
- **iwconfig wlan0**
- The output should show Mode: Managed if in managed mode.
- **Switching to Managed Mode:**
- **sudo ifconfig wlan0 down**
- **sudo iwconfig wlan0 mode managed**
- **sudo ifconfig wlan0 up**

2. Monitor Mode

- Definition: A mode where the wireless adapter captures all wireless traffic in the vicinity, regardless of the destination.

Use Case: Essential for tasks such as packet sniffing, network analysis, and penetration testing.

Commands:

- **Enable Monitor Mode:**
- **sudo airmon-ng start wlan0**
- Disable Monitor Mode:
- **sudo airmon-ng stop wlan0mon**
- Check Current Mode:
- **iwconfig wlan0mon**
- The output should show Mode: Monitor.

3. Switching Between Modes

- It's crucial to switch back to managed mode after completing penetration testing, as monitor mode doesn't allow regular network communications.
- Switching Back:
- **sudo ifconfig wlan0mon down**
- **sudo iwconfig wlan0mon mode managed**
- **sudo ifconfig wlan0mon up**

Section 1: Understanding Wi-Fi Networks

1.1 What is a Wi-Fi Network?

- **Explanation**: Wi-Fi networks are wireless networks that use radio waves to connect devices to the internet or each other without physical cables.
- **Example**: When you connect your smartphone to your home's Wi-Fi, you're using a Wi-Fi network.

1.2 Wi-Fi Bands: 2.4 GHz and 5 GHz

- **Explanation**: Wi-Fi networks operate on two main frequency bands:
- **2.4 GHz**: Longer range, slower speed, more interference.
- **5 GHz**: Shorter range, faster speed, less interference.
- **Example**: Your home router might support both bands, and you can choose which one to connect to based on your distance from the router.

1.3 Wi-Fi Security Protocols

- **Explanation**:
- **WEP (Wired Equivalent Privacy)**: An outdated and easily crackable protocol.
- **WPA (Wi-Fi Protected Access)**: More secure than WEP, but still vulnerable.
- **WPA2 (Wi-Fi Protected Access II)**: The most common and secure, but not invulnerable.
- **WPA3**: The latest and most secure protocol.
- **Example**: When setting up your router, you may choose WPA2 for the best balance of security and compatibility.

1.4 Basic Terminology

- **BSSID (Basic Service Set Identifier)**: The MAC address of a wireless access point.
- **SSID (Service Set Identifier)**: The name of the Wi-Fi network.
- **MAC Address**: A unique identifier assigned to network interfaces.
- **Handshake**: A process where devices authenticate and establish a connection.
- **Packet**: A unit of data transmitted over a network.

Section 2: Preparing for Penetration Testing

2.1 Legal and Ethical Considerations

- **Explanation**: Always obtain permission before testing any network that you do not own. Unauthorized access to networks is illegal and unethical.
- **Example**: Before testing a client's network, ensure you have a signed agreement outlining the scope of your testing.

2.2 Setting Up Your Environment

- **Explanation**: You need a computer with Wi-Fi capabilities, preferably with a wireless adapter that supports monitor mode. Kali Linux is a popular operating system for penetration testing.
- **Commands**:
- **ifconfig** — Displays network interfaces.
- **iwconfig** — Displays wireless network interfaces and settings.
- **airmon-ng** — Enables monitor mode on wireless interfaces.

2.3 Installing Necessary Tools

- **Tools**:
- **Aircrack-ng**: A suite of tools for Wi-Fi network auditing.
- **Wireshark**: A network protocol analyzer.
- **Reaver**: A tool for brute-force attacking WPS.
- **Commands**:
- sudo apt-get install aircrack-ng wireshark reaver — Installs the necessary tools on Kali Linux.

Section 3: Reconnaissance and Information Gathering

3.1 Scanning for Wi-Fi Networks

- **Explanation**: The first step is to identify the networks around you and gather information such as BSSID, SSID, channel, and encryption type.
- **Commands**:
- **airmon-ng start wlan0** — Puts the wireless interface into monitor mode.
- **airodump-ng wlan0mon** — Scans and displays nearby Wi-Fi networks.
- **Example**: You run airodump-ng wlan0mon and see a list of Wi-Fi networks along with their BSSID, SSID, and encryption types.

3.2 Targeting a Specific Network

- **Explanation**: Once you've identified your target network, focus on gathering detailed information about it.
- **Commands**:
- **airodump-ng — bssid [BSSID] -c [channel] -w capture wlan0mon** — Captures packets from a specific network.
- **Example**: You find a network with weak encryption and start capturing packets to analyze later.

Section 4: Pre-Connection Attacks

4.1 Packet Sniffing

- **Explanation**: Packet sniffing involves capturing and analyzing the data packets transmitted over the network to find vulnerabilities.
- **Commands**:
- **airodump-ng — bssid [BSSID] -c [channel] -w capture wlan0mon** — Captures packets from a specific network.
- wireshark — Analyzes captured packets.
- **Example**: You capture packets from a target network and use Wireshark to inspect them for any sensitive information.

4.2 Deauthentication Attack

- **Explanation**: This attack forces devices off the network, causing them to reconnect. This can be used to capture handshakes or disrupt the network.
- **Commands**:
- **aireplay-ng — deauth 0 -a [BSSID] wlan0mon** — Sends deauthentication packets to a network.
- **Example**: You force a device to disconnect from the network, and when it reconnects, you capture the handshake.

Section 5: Gaining Access — Cracking WEP

5.1 Introduction to WEP Cracking

- **Explanation**: WEP is an outdated and insecure encryption protocol. Cracking WEP involves capturing enough packets to guess the key.
- **Commands**:
- **airodump-ng — bssid [BSSID] -c [channel] -w capture wlan0mon** — Captures packets.
- **aircrack-ng capture*.cap** — Attempts to crack the WEP key using captured packets.
- **Example**: After capturing enough packets, you successfully crack the WEP key and gain access to the network.

5.2 Fake Authentication Attack

- **Explanation**: Before cracking WEP, you may need to authenticate with the network using a fake request.
- **Commands**:
- **aireplay-ng -1 0 -e [SSID] -a [BSSID] -h [MAC] wlan0mon** — Sends a fake authentication request.
- **Example**: You send a fake authentication request to the network to start capturing packets.

5.3 ARP Request Replay Attack

- **Explanation**: This attack generates more traffic on the network by replaying ARP requests, speeding up the process of capturing packets for WEP cracking.
- **Commands**:
- **aireplay-ng -3 -b [BSSID] -h [MAC] wlan0mon** — Replays ARP requests.
- **Example**: You increase network traffic to quickly capture enough packets for cracking WEP.

Section 6: Gaining Access — Cracking WPA/WPA2

6.1 Introduction to WPA/WPA2 Cracking

- **Explanation**: WPA and WPA2 are more secure than WEP, but they can still be cracked by capturing handshakes and using brute-force attacks with a wordlist.
- **Commands**:
- **airodump-ng — bssid [BSSID] -c [channel] -w capture wlan0mon** — Captures handshakes.
- **aircrack-ng -w wordlist.txt -b [BSSID] capture*.cap** — Cracks WPA/WPA2 using a wordlist.
- **Example**: You capture a WPA handshake and use a wordlist to crack the password.

6.2 Capturing the WPA/WPA2 Handshake

- **Explanation**: Capturing a handshake is essential for cracking WPA/WPA2. The handshake occurs when a device connects to the network.
- **Commands**:
- **airodump-ng — bssid [BSSID] -c [channel] -w capture wlan0mon** — Captures handshakes.
- **Example**: You capture a handshake when a device reconnects after a deauthentication attack.

6.3 Cracking WPA/WPA2 with a Wordlist

- **Explanation**: After capturing the handshake, use a wordlist to try various passwords until one matches.
- **Commands**:
- **aircrack-ng -w wordlist.txt -b [BSSID] capture*.cap** — Cracks WPA/WPA2 using a wordlist.
- **Example**: You use a popular wordlist like “rockyou.txt” to crack the WPA2 password.

6.4 Hacking WPA/WPA2 Without a Wordlist

- **Explanation**: If you don’t have a wordlist, you can attempt other methods, such as exploiting weak WPS (Wi-Fi Protected Setup).
- **Commands**:
- **reaver -i wlan0mon -b [BSSID] -vv** — Uses a brute-force attack on WPS to retrieve the WPA/WPA2 passphrase.

- **Example**: You exploit a vulnerable WPS implementation to gain access to a WPA2 network without a wordlist.

Section 7: Post-Connection Attacks

7.1 Man-in-the-Middle (MITM) Attack

- **Explanation**: A MITM attack involves intercepting and potentially altering communication between a device and the network.
- **Commands**:
- **ettercap -T -q -i wlan0mon -M arp:remote /[victim_IP]/ /[router_IP]/** — Launches an ARP poisoning MITM attack.

Example: You intercept traffic between a victim's device and the router, capturing sensitive data.

7.2 DNS Spoofing

- **Explanation**: DNS spoofing involves redirecting the victim's traffic to a fake website instead of the intended one.
- **Commands**:
- **ettercap -T -q -i wlan0mon -M arp:remote /[victim_IP]/ /[router_IP]/ -P dns_spoof** — Spoofs DNS requests.
- **Example**: You redirect a victim trying to visit www.bank.com to your own phishing page.

7.3 Session Hijacking

- **Explanation**: After a MITM attack, you can hijack active sessions by stealing cookies or session tokens.
- **Commands**:
- ferret -i wlan0mon — Captures session information.
- hamster — Replays captured sessions.
- **Example**: You hijack a victim's session on a web application by stealing their session cookies.

Section 8: Securing Wi-Fi Networks

8.1 Changing Default Settings

- **Explanation**: Always change default router settings such as SSID, admin username, and password.
- **Example**: Changing the default SSID from "TP-Link_1234" to something unique.

8.2 Using Strong Encryption

- **Explanation**: Use WPA3 or WPA2 with a strong passphrase and disable WPS.

- **Example**: Setting a passphrase like "S3curePa$$w0rd!@#" and disabling WPS.

8.3 Enabling MAC Address Filtering

- **Explanation**: Limit network access to specific MAC addresses.
- **Example**: Adding the MAC addresses of your devices to the router's whitelist.

8.4 Regularly Updating Firmware

- **Explanation**: Keep your router's firmware up to date to patch known vulnerabilities.
- **Example**: Checking for and applying firmware updates every few months.

8.5 Monitoring Network Activity

- **Explanation**: Regularly monitor your network for any unusual activity or unknown devices.
- **Example**: Using network monitoring tools like Fing to see all devices connected to your Wi-Fi.

Tools Use For Network Penetration Testing

Nmap Commands

Nmap is a powerful tool for network discovery and security auditing.

Basic Commands:

1. **Ping Scan (Find Live Hosts):**

- Command: nmap -sn 192.168.1.0/24
- **Use Case**: Discover which hosts are up in a subnet.
- **Example**: Scan your local network (e.g., 192.168.1.0/24) to see which devices are online.

1. **Scan for Open Ports:**

- Command: nmap -p 80,443 192.168.1.5
- **Use Case**: Check if a specific host has ports 80 and 443 open.
- **Example**: See if a web server is running on a host.

1. **Service Version Detection:**

- Command: nmap -sV 192.168.1.5
- **Use Case**: Identify what services and versions are running on open ports.
- **Example**: Determine which web server (e.g., Apache, Nginx) is running on a host.

1. **Aggressive Scan:**

- Command: nmap -A 192.168.1.5
- **Use Case**: Perform a comprehensive scan including OS detection, version detection, script scanning, and traceroute.
- **Example**: Gather detailed information about a target host.

1. **Scan a Range of IPs:**

- Command: nmap 192.168.1.1–254
- **Use Case**: Scan all hosts in a specific IP range.
- **Example**: Discover all devices within a given subnet.

1. **Scan Specific Ports:**

- Command: nmap -p 22,80,443 192.168.1.1
- **Use Case**: Check for specific services running on a host.
- **Example**: Verify if SSH, HTTP, and HTTPS are enabled on a server.

1. **Scan with Stealth Mode (SYN Scan):**

- Command: nmap -sS 192.168.1.5
- **Use Case**: Perform a stealth scan that is less likely to be detected by firewalls or IDS.
- **Example**: Test the security of a host without triggering alarms.

1. **Save Scan Results to a File:**

- Command: nmap -oN scan_results.txt 192.168.1.5
- **Use Case**: Save the output of a scan to a file for documentation.
- **Example**: Store scan results for later analysis.

Hydra Commands

Hydra is a tool used for brute-force password attacks on various services.

Basic Commands:

1. **Brute-Force FTP Login:**

- Command: hydra -l admin -P passwords.txt ftp://192.168.1.5
- **Use Case**: Attempt to brute-force the admin account on an FTP server.
- **Example**: Test the security of an FTP server by trying multiple passwords.

1. **Brute-Force SSH Login:**

- Command: hydra -l root -P passwords.txt ssh://192.168.1.5
- **Use Case**: Attempt to brute-force the root account on an SSH server.
- **Example**: Test the strength of SSH passwords on a server.

1. **Brute-Force HTTP-POST Form:**

- Command: hydra -l admin -P passwords.txt 192.168.1.5 http-post-form "/login.php:user=^USER^&pass=^PASS^:F=incorrect"
- **Use Case**: Test the login form of a web application.
- **Example**: Perform a brute-force attack on a web login page.

1. **Parallel Connections:**

- Command: hydra -t 16 -l admin -P passwords.txt ssh://192.168.1.5
- **Use Case**: Increase the speed of the attack by using 16 parallel connections.
- **Example**: Speed up a brute-force attack by increasing the number of attempts made simultaneously.

1. **Verbose Mode (Show Each Attempt):**

- Command: hydra -V -l admin -P passwords.txt ssh://192.168.1.5
- **Use Case**: Display each login attempt in real-time.
- **Example**: Monitor the progress of a brute-force attack as it happens.

Nikto Commands

Nikto is a web server scanner that identifies vulnerabilities.

Basic Commands:

1. **Scan a Website:**

- Command: nikto -h http://192.168.1.5
- **Use Case**: Scan a web server for known vulnerabilities.
- **Example**: Identify potential security issues on a web server.

1. **Scan with SSL:**

- Command: nikto -h https://192.168.1.5
- **Use Case**: Scan a web server using HTTPS.
- **Example**: Test a secure web server for vulnerabilities.

1. **Save Scan Results to a File:**

- Command: nikto -h http://192.168.1.5 -o results.txt
- **Use Case**: Save the output of the scan to a text file.
- **Example**: Document the results of a web server scan for future reference.

1. **Scan Multiple Ports:**

- Command: nikto -h http://192.168.1.5 -p 80,443,8080
- **Use Case**: Scan a web server on multiple ports.
- **Example**: Check for vulnerabilities on all web services running on a server.

1. **Scan Specific Directories:**

- Command: nikto -h http://192.168.1.5 -Tuning 2
- **Use Case**: Focus the scan on interesting directories and files.
- **Example**: Target sensitive directories like /admin or /config.

1. **Run Nikto in Quiet Mode:**

- Command: nikto -h http://192.168.1.5 -quiet
- **Use Case**: Run the scan without verbose output.
- **Example**: Perform a quiet scan with minimal output.

Snort Commands

Snort is an intrusion detection system (IDS) that monitors network traffic for suspicious activity.

Basic Commands:

1. **Run Snort in IDS Mode:**

- Command: snort -A console -q -c /etc/snort/snort.conf
- **Use Case**: Monitor network traffic and log any suspicious activity.
- **Example**: Detect potential intrusions on your network.

1. **Run Snort in Packet Logging Mode:**

- Command: snort -dev -l /var/log/snort
- **Use Case**: Capture and log all network traffic.
- **Example**: Store network packets for later analysis.

1. **Test Snort Configuration:**

- Command: snort -T -c /etc/snort/snort.conf
- **Use Case**: Test the Snort configuration file for errors.
- **Example**: Ensure Snort is properly configured before deploying it in a production environment.

1. **Run Snort with Specific Interface:**

- Command: snort -i eth0 -c /etc/snort/snort.conf
- **Use Case**: Monitor a specific network interface.
- **Example**: Focus Snort's monitoring on the eth0 interface.

1. **Run Snort in Daemon Mode:**

- Command: `sn