COMP4337/9337 Securing Fixed and Wireless Network
Lab 3: Man in the Middle Attack and Evil Twin Access Point

Objectives

This lab is designed to help the students to learn:

- how to exploit the vulnerability which allows an unauthenticated, adjacent attacker to terminate a valid user connection at layer 2 and to hijack the user at layer 3 and
- how to launch an MITM attack by exploiting no encryption vulnerability.

Lab Overview

A Man in the Middle Attack (MITM) is a cyberattack where the attacker secretly relays and possibly alters the communications between two parties who believe that they are directly communicating with each other. To simulate an MITM attack, we will be using the following devices:

- a headless Raspberry Pi controlled via SSH (as a rogue access point) and
- your own Android or iOS device (as the victim)

For the software, we will use a tool named `mitmproxy`, which is an open-source proxy application that allows intercepting HTTP and HTTPS connections between any HTTP(S) client (such as a mobile or desktop browser) and a web server using a typical MITM attack scenario. Like other proxies (such as Squid[1]), it accepts connections from clients and forwards them to the destination server. However, while other proxies typically focus on content filtering or speed optimization through caching, the goal of `mitmproxy` is to let an attacker monitor, capture and alter these connections in real-time.

Assessment and Marking

This lab consists of two parts:

- Part A: Man in the Middle Attack (hands-on)
- Part B: Evil Twin Access Point (AP) (conceptual/theoretical)

Part A involves hands-on activities about the MITM attack, while Part B only requires you to conceptually explain the Evil Twin AP. You must complete **Part A by end of your lab**. A separate **Lab 3 Assessment** will be available on **Moodle**. Submission deadline is **Monday 14th March 2022, 23:59h**.

The marks will be made available on Moodle within 2 weeks of the submission date. The details of the marks are as follows:

- Total marks and weight for Lab 3 are 100 and 0.2/1, respectively.
    - Lab 3 Performance (25 marks)
    - Lab 3 Assessment, submission on Teams (75 marks)

[1] http://www.squid-cache.org/

- Students who do not attend the lab will **lose ALL 100 marks.**
- **Every student must show their work for Part A irrespective of being in a group, unless there is a valid excuse in order to get the Lab performance marks.**

Note: Lab performance involves tutor asking questions, feedback, and comments about the activity while the lab is in progress. Hence, if a group is found to be cheating or submitting a work that does not match what the tutor observes to be the team's performance, then **NO MARK** will be awarded for Part A.

- The standard late penalty introduced under UNSW new assessment implementation procedure will be applied for this course.
    - 5% per day,
    - for all assessments where a penalty applies,
    - capped at five days (120 hours) from the assessment deadline, after which a student cannot submit an assessment, and
    - no permitted variation.

Part A: Man in the Middle Attack

The network topology for our MITM attack scenario is depicted in the following figure.

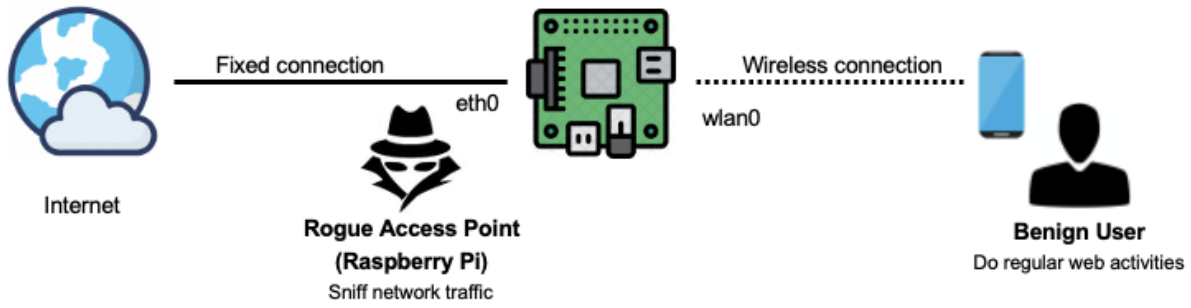


Figure 1. Our network topology.

Note: As the Raspberry Pi is not connected to the internet via a fixed connection, we will be simulating a web server that runs locally on the Raspberry Pi itself. If you have a fixed connection at home, you may connect the Raspberry Pi to your home router via an ethernet cable.

Please follow these instructions to complete the lab:

1. We need to setup transparent proxy-ing to forward network packets. To set up transparent proxy-ing, we need two new components. The first is a redirection mechanism that transparently reroutes a TCP connection destined for a server on the Internet to a listening proxy server. This usually takes the form of a firewall on the same host as the proxy server, i.e., iptables on Linux. When the proxy receives a redirected connection, it sees a vanilla HTTP request, without a host specification. This is where the second new component comes in, i.e., a host module that allows us to query the redirector for the original destination of the TCP connection.
2. Connect to the Raspberry Pi and get a remote shell, either by using SSH or by opening a new Terminal window on VNC. Please refer to Prep 2 docs for detailed instruction.

3. On your remote shell, type the following commands to the terminal to setup transparent proxy-ing:

```
$ sudo sysctl -w net.ipv4.ip_forward=1
```

We need to route any traffic of HTTP (port 80) and HTTPS (443) to `mitmproxy`, which will be listening to port 8080:

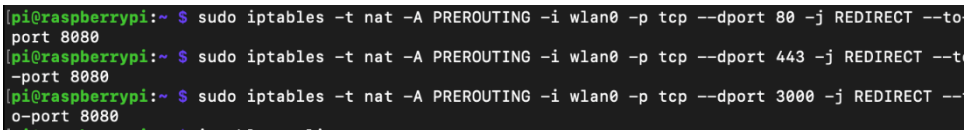
```
$ sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 80
  -j REDIRECT --to-port 8080
```

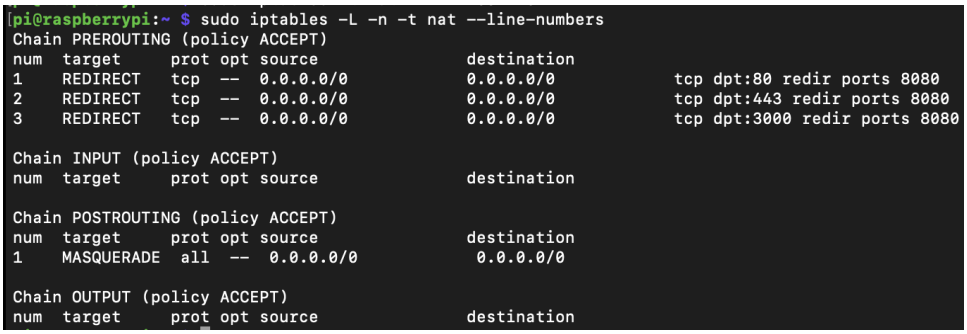
```
$ sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 443
      -j REDIRECT --to-port 8080
```

4. We will also need to route traffic to our local web server on port 3000 to `mitmproxy`:

```
$ sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 3000

  -j REDIRECT --to-port 8080
```





5. We have provided a script for the local web server in the Raspberry Pi where the user may input their login credentials. Run the web server by moving to directory `~/sfwn/login/` and start the server using `login.js`.

```
$ cd ~/sfwn/login/
```

```
$ node login.js
```

Leave the command running, you may see a blank output.

6. Open another remote shell to the Raspberry Pi and run `mitmproxy`, which has been installed on our custom Raspberry Pi OS.

```
$  mitmproxy --mode transparent --showhost
```

Keep the terminal window open and do not close it.

7. On your Android/iOS device, connect to `testnetwork` Access Point and type `comp43379337` for password when prompted.
8. Open a web browser on your Android/iOS device and type `login-page.com:3000` on the address bar to access our local web server. Also include the port number (3000). Now try to login using your zID as username and "kali" as password.

   Note what happened. You will not be able to login as our local server only contains a dummy login page, which is normally set by hackers to capture login credentials.

   Note: We have set up a local DNS record that would point `login-page.com` to our local server. If your browser is not directing you to the correct login page, try to type the address manually, i.e., `192.168.4.1:3000`.

9. Return to your Raspberry Pi and revisit `mitmproxy` terminal window. You will see a list of HTTP messages.
10. To filter out intercepted packets, set view filter to `auth` by pressing `'f'` followed by `auth`, then press ENTER.
11. Browse through POST requests using up and down arrow keys. Press ENTER to view the request in detail and press `'q'` to return to the list. Try to find the POST request containing the username and password you have provided for `login-page.com` previously.
12. **Important!** Ask your tutor to verify that you have correctly located the username and password exchanged using HTTP. You may want to share your screen or provide a screenshot.

You just extracted username and passwords exchanged over an HTTP connection!

Part B: Evil Twin Access Point

You may want to read the Appendix to familiarise with the concept of Evil Twin Attack. The network topology for our Evil Twin AP attack scenario is depicted in the following figure.

Figure 2. Evil Twin AP topology.

In Part A, we used the Raspberry Pi as our rogue AP. As you may have noticed this is easily noticeable as it is hard to convince our target (victim) to connect to the rogue AP.

In Part B, your task is to find a way to force the victim, that has been connected to a legitimate AP, to get connected to our rogue AP. Note that the rogue AP must also route the user traffic to Internet so the victim would not notice connection to fake AP. You will also need to disconnect clients connected to the real AP, so on reconnecting they will be associated to your rogue AP.

Consider the following AP available on a public location:

AP name: Starbucks
Channel: 2
Encryption: None

Explain the process to setup this working Evil Twin AP in **Lab 3 Assessment on Teams!**

Appendix: Wireless Hijacking

The following describes the concept of wireless hijacking, in which the attacker de-authenticate the victim to force the victim to get connected to the rogue AP. Prior to the attack the attacker configures access point on his laptop with client-card, with the same SSID as that of the Target AP ("blue" in above example). This way attacker's access point is now functioning as an evil twin AP of the Target AP. Note that, the evil twin will be transmitting on a different channel than the public hotspot (channel 11 vs channel 6 in the figure).

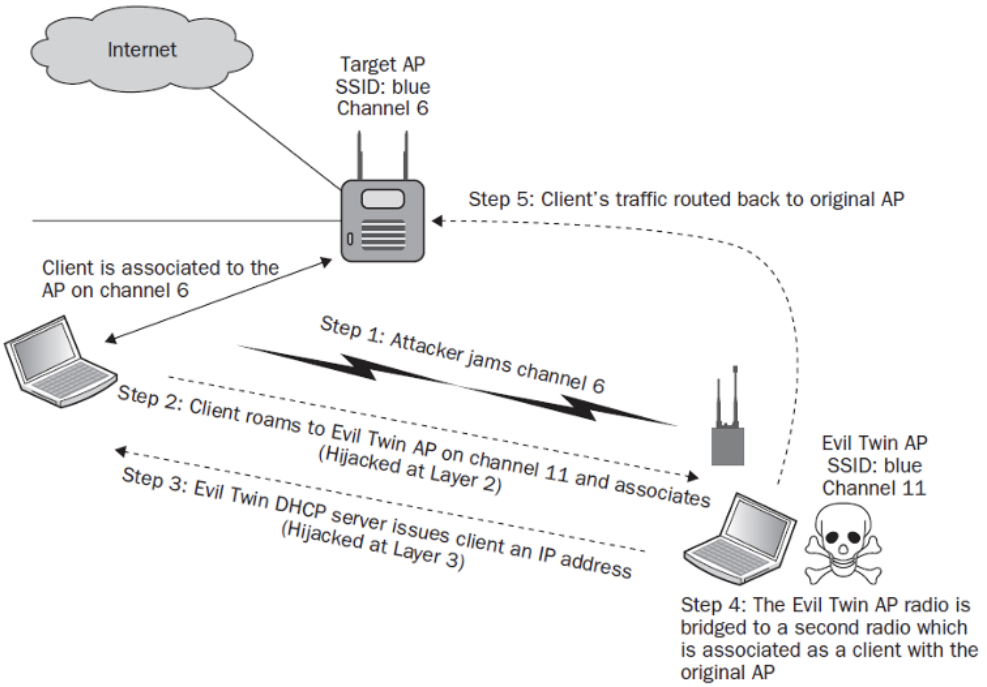


Figure 3. Wireless Hijacking/MITM Attack (Source: CWSP Book, Page: 318).

1. **Step 1:** RF jammers can also be used to force any clients to roam to an evil twin AP. Note that de-authentication frames are usually used as one way to start a hijacking attack, instead of jamming the channel.
2. **Step 2:** In order to force clients to leave the Target AP and join this new evil twin, attacker then sends spoofed disassociation or de-authentication frames, forcing client stations associated with the Target AP to roam to the evil twin access point. Note that in this step the attacker has hijacked the client stations at Layer 2.
3. **Step 3:** The evil twin AP will typically be configured with a Dynamic Host Configuration Protocol (DHCP) server available to issue IP addresses to the clients. The user's computer could, during the process of connecting to the evil twin, fall victim to the DHCP attack (An attack that exploits the DHCP process to dump root kits or other malware onto the victim's computer in addition to giving them an IP address). Note that here the attacker has hijacked the client stations at Layer 3.
4. **Step 4:** The attacker may also be using a second wireless card with their laptop to execute what is known as a man-in-the-middle attack, as shown in the figure. The second WLAN card is associated with the original access point as a client. The attacker has bridged together their second wireless card with the Wi-Fi card that is being used as the evil twin access point.

5. **Step 5:** The traffic from the client is now routed from the evil twin access point through the second Wi-Fi card, right back to the original access point from which the users have just been hijacked.
6. The result is that the users remain hijacked; however, they still have a route back through the gateway to their original network, so they never know they have been hijacked. The attacker can therefore sit in the middle and execute peer-to-peer attacks indefinitely while remaining completely unnoticed.