Lab: Investigating WiFi Connections with ESP32 and Wireshark

Overview

In this lab, you will use an ESP32 development board to connect to a WPA2-protected WiFi network and observe the detailed steps involved in the connection process, including the 4-way handshake. You will monitor WiFi events in your code and capture traffic with Wireshark to correlate physical and logical behaviors.

Objectives

- Understand the steps involved in establishing a WiFi connection
- Observe WPA2's 4-way handshake using Wireshark
- Identify and explain key WiFi frame types
- Reflect on how security is enforced during the connection process

Prerequisites

- ESP32 dev board
- Arduino IDE or PlatformIO installed
- Wireshark installed on a laptop or second device
- A WPA2-capable WiFi network (router or mobile hotspot)
- (Optional) A separate device capable of capturing WiFi in monitor mode

Part 1: ESP32 Code

Write a sketch that connects to your network and logs WiFi events.

```
#include <WiFi.h>

const char* ssid = "YourNetwork";
const char* password = "YourPassword";

void onWiFiEvent(WiFiEvent_t event) {
  switch (event) {
    case SYSTEM_EVENT_STA_START:
      Serial.println("WiFi started, connecting...");
      break;
    case SYSTEM_EVENT_STA_CONNECTED:
      Serial.println("Connected to AP, awaiting IP...");
      break;
    case SYSTEM_EVENT_STA_GOT_IP:
      Serial.print("Got IP: ");
      Serial.println(WiFi.localIP());
      break;
    case SYSTEM_EVENT_STA_DISCONNECTED:
      Serial.println("Disconnected from WiFi");
      break;
    default:
      break;
  }
}

void setup() {
  Serial.begin(115200);
  WiFi.onEvent(onWiFiEvent);
  WiFi.begin(ssid, password);
}

void loop() {
  delay(1000);
}
```

Part 2: WiFi Packet Capture with Wireshark

1. Power off the ESP32.
2. Open Wireshark and begin capturing on the correct WiFi channel.
3. Apply the following display filter to limit view to authentication traffic:

```
eapol || wlan.fc.type_subtype == 0x00 || wlan.fc.type_subtype == 0x01
```

1. Power on the ESP32.
2. Observe and record the following in Wireshark:
3. Authentication Request/Response
4. Association Request/Response
5. 4 EAPOL frames (the handshake)

6. DHCP traffic (optional)
7. Save the capture to a file.

Part 3: Reflection Questions

Answer the following in your report:

1. What types of WiFi frames were visible during the connection process?
2. At what point did the ESP32 receive its IP address?
3. What is the purpose of each of the 4 EAPOL handshake messages?
4. What would happen if message 3 was retransmitted multiple times?
5. How could you detect a rogue access point or deauthentication attack?

Bonus Challenges (Optional)

- Change the SSID or password and observe ESP32 behavior and logs.
- Try programming the ESP32 to operate in promiscuous mode to detect deauth packets.
- Configure the ESP32 as a soft AP and capture the handshake from a connecting device.

Submission

Submit the following:

- A copy of your ESP32 sketch
- A screenshot or `.pcap` showing the handshake process
- Your reflection answers in a short report (½ page)
- (Optional) any additional findings or bonus challenge results