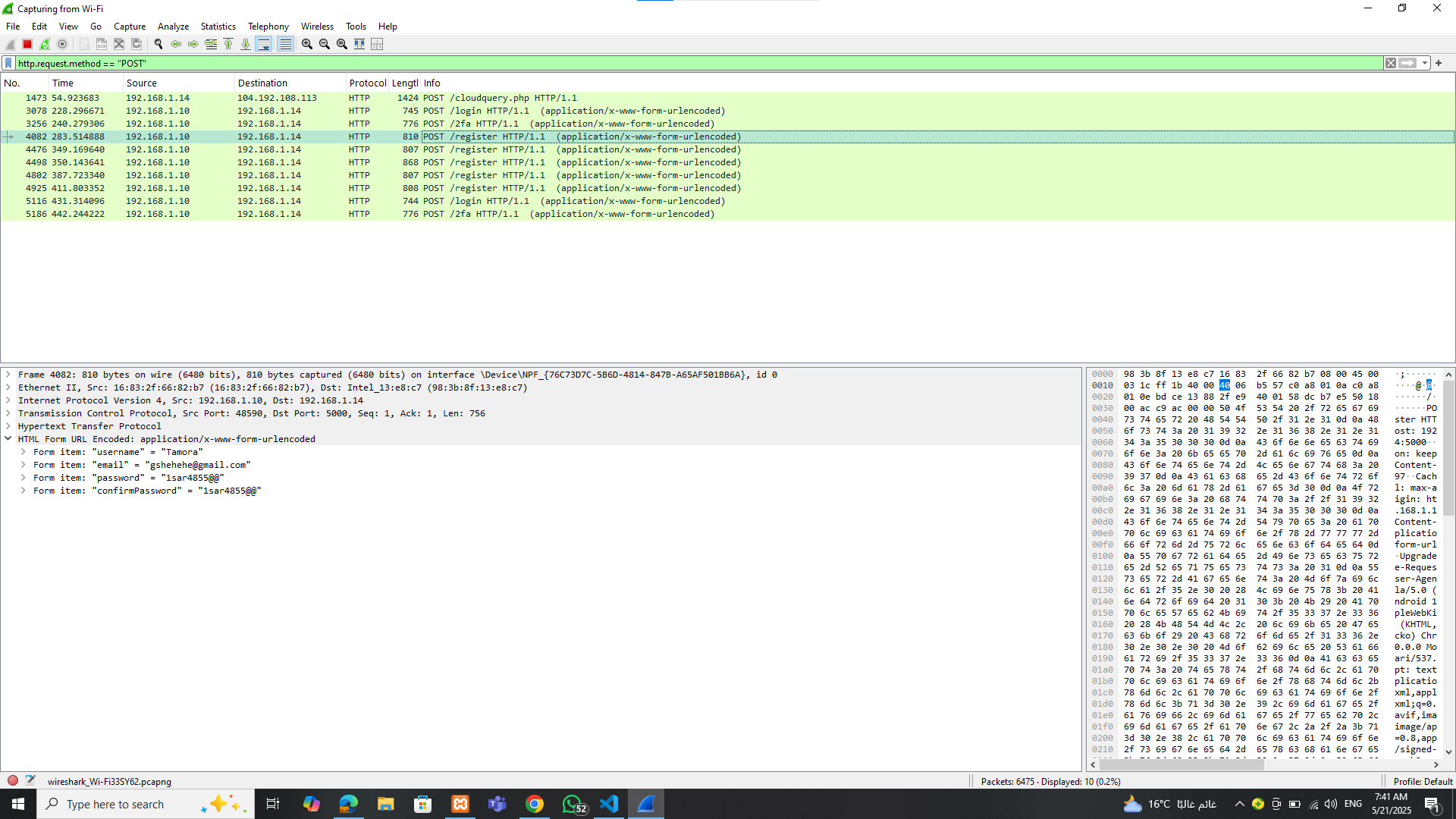
**Frist: HTTP:**

This screenshot shows a successful Man-in-the-Middle (MITM) attack over HTTP:



Wireshark captured the login credentials sent in plain text:

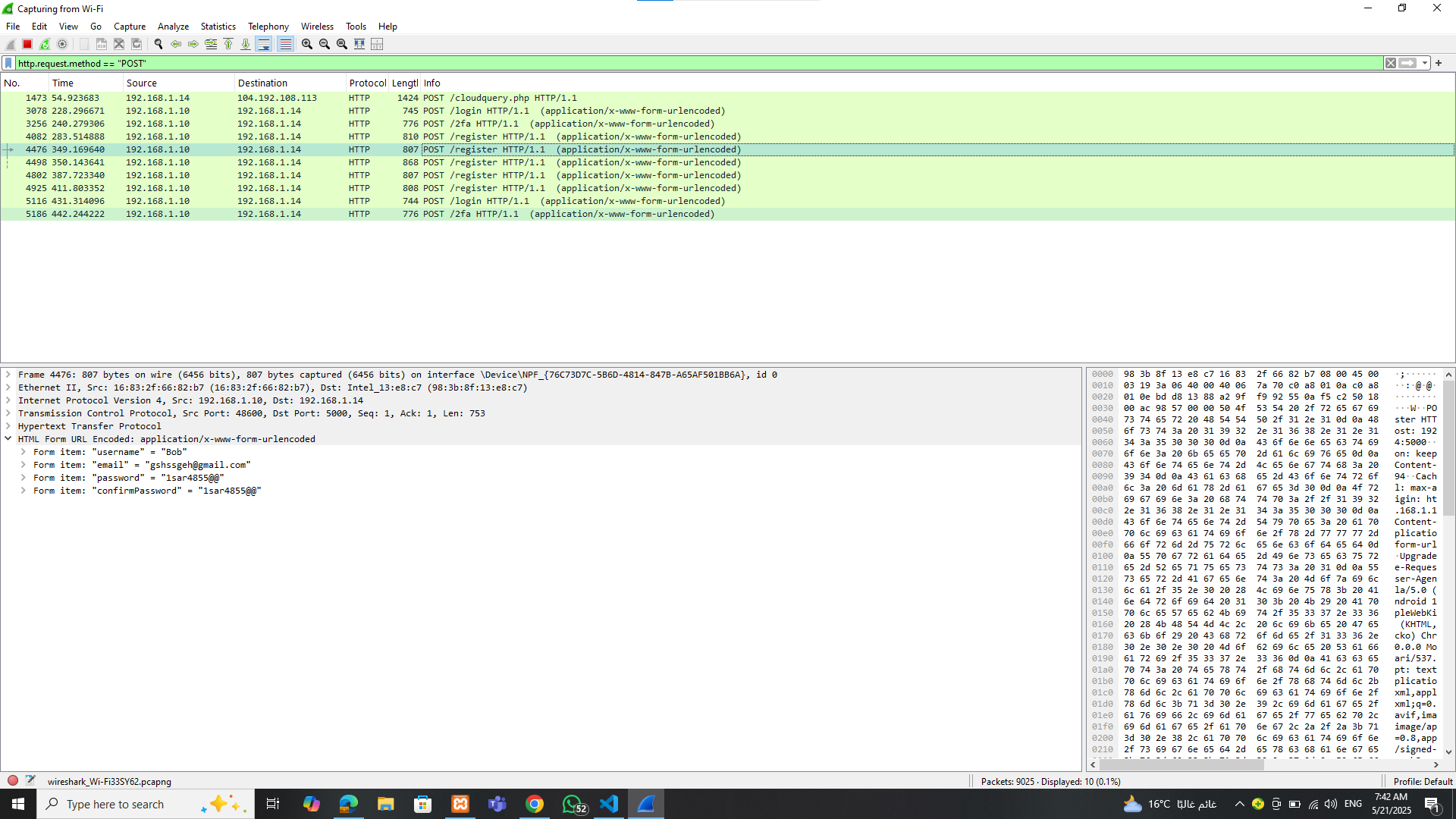
username=tamora

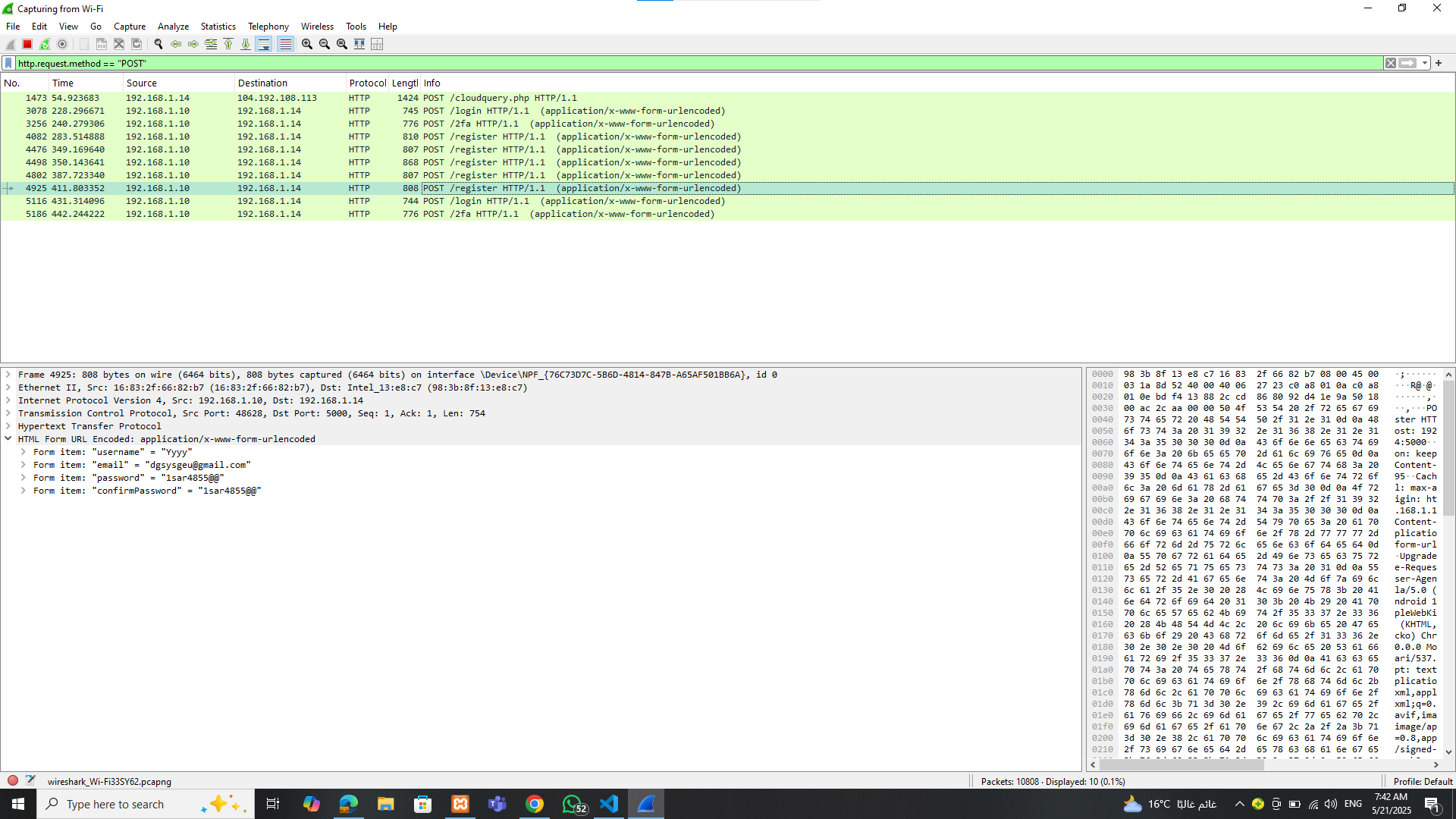
password=1sar4855@@.

**Explanation:**

* The communication between client and server is unencrypted.
* Any attacker on the same network can intercept and read sensitive data.
* This proves that HTTP is not secure for handling authentication

**Examples with difference user name:**





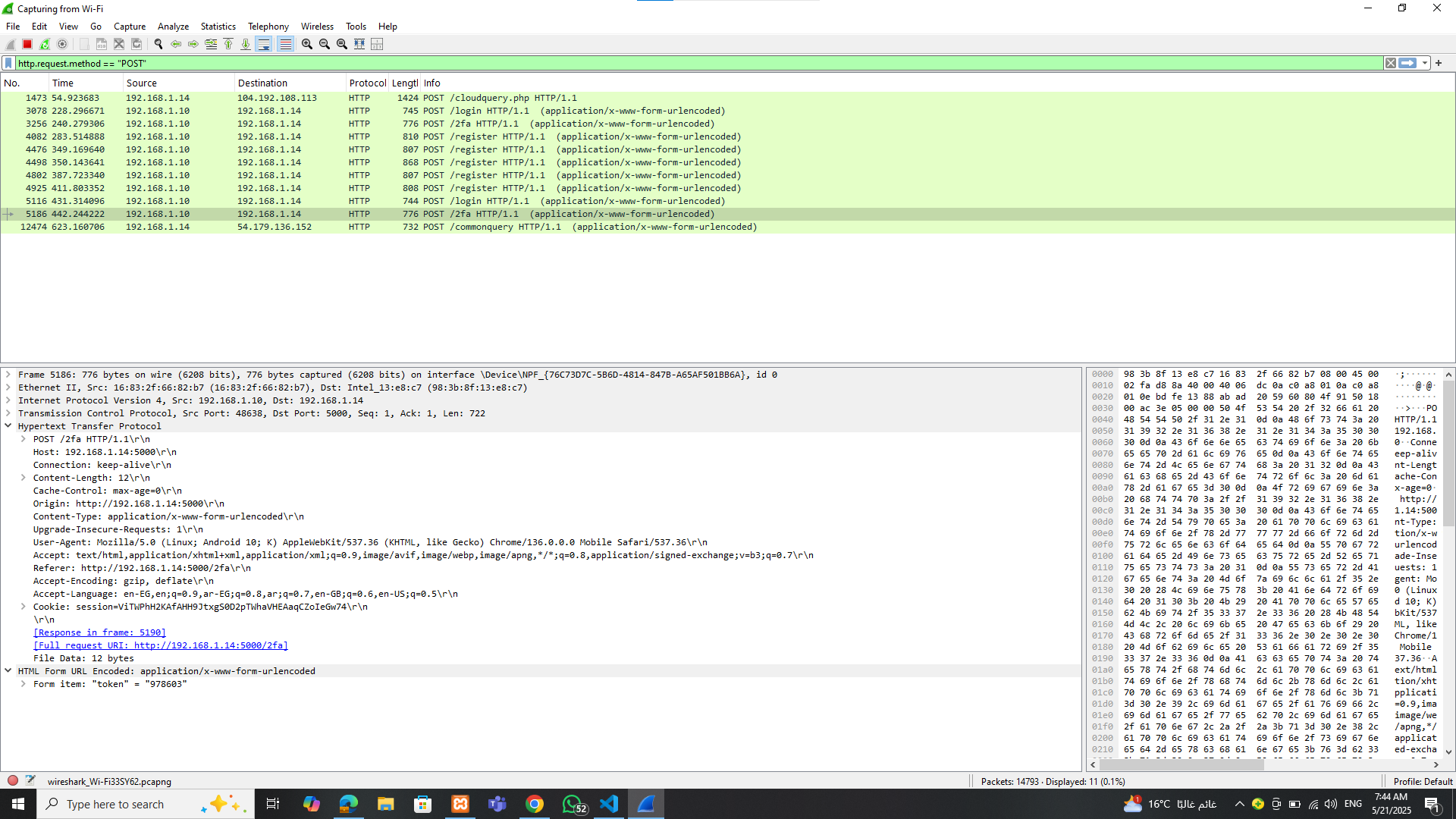
**Two-Factor Authentication:**

This screenshot shows the Two-Factor Authentication (2FA) step in the login process.

After the username and password are verified, the user is required to enter a 6-digit verification code.

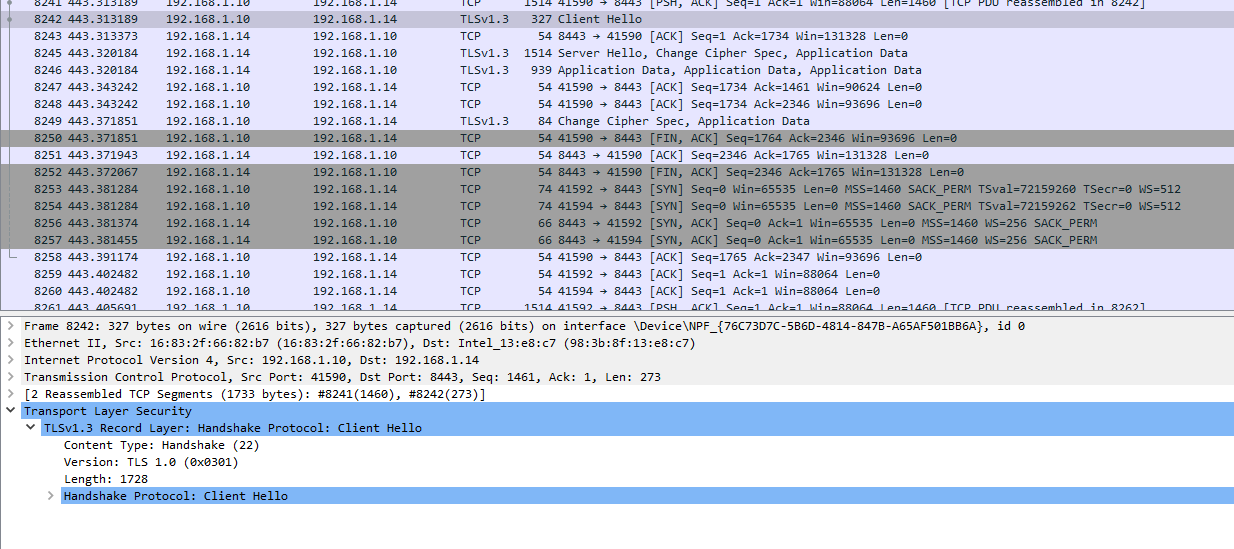
This additional layer of security protects user accounts even if login credentials are compromised.

It ensures that only the legitimate user, with access to the authenticator app or email, can complete the login process.



**2- HTTPS:**

This screenshot shows the same login request sent over HTTPS

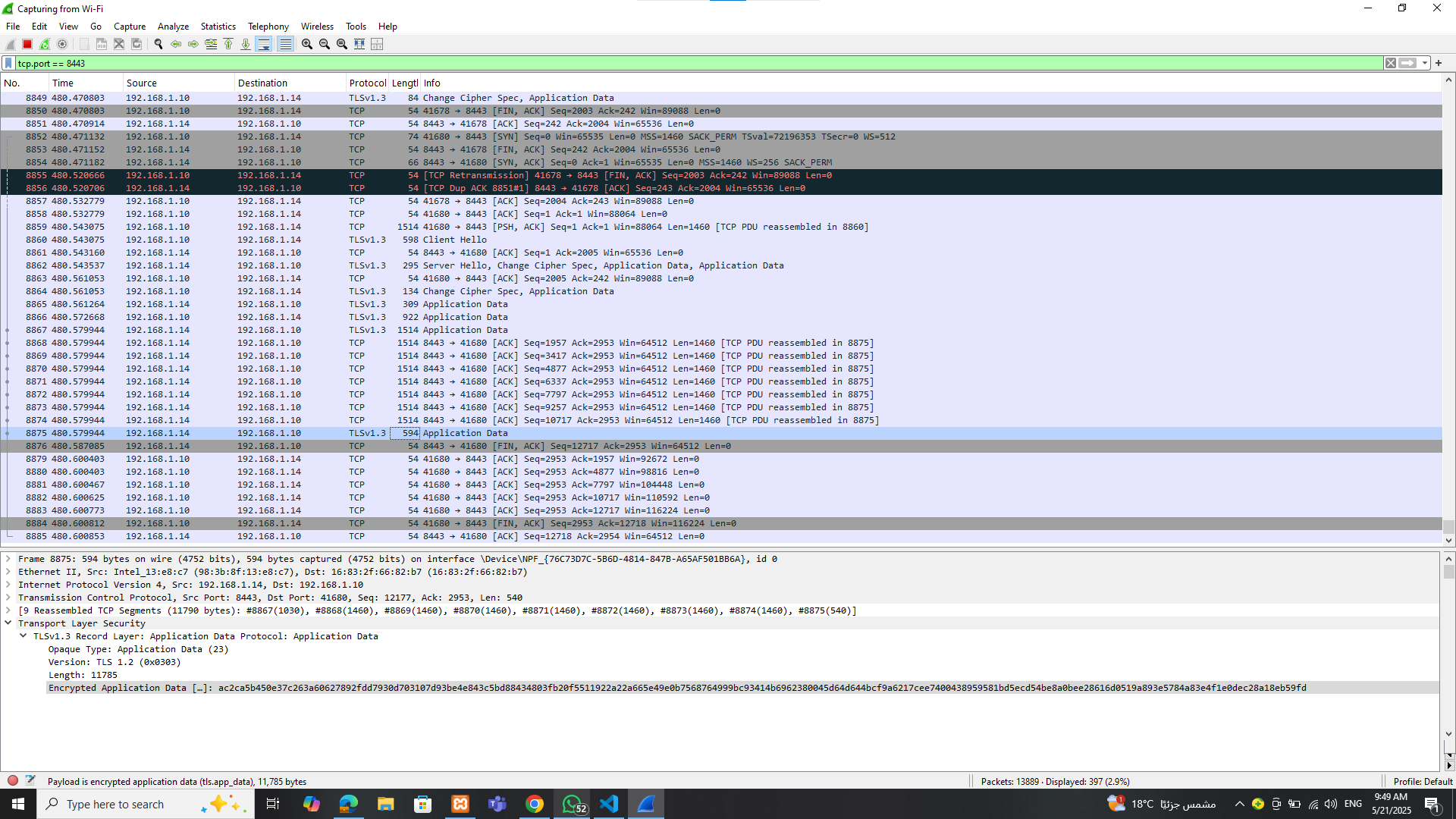
Wireshark does not display the credentials. Instead, it shows

**Explanation:**

* The TLS handshake begins with "Client Hello".
* No user data or password appears in the traffic.
* This confirms the connection is encrypted and secure.

**Encrypted Data in HTTPS:**

This screenshot shows encrypted traffic under HTTPS:



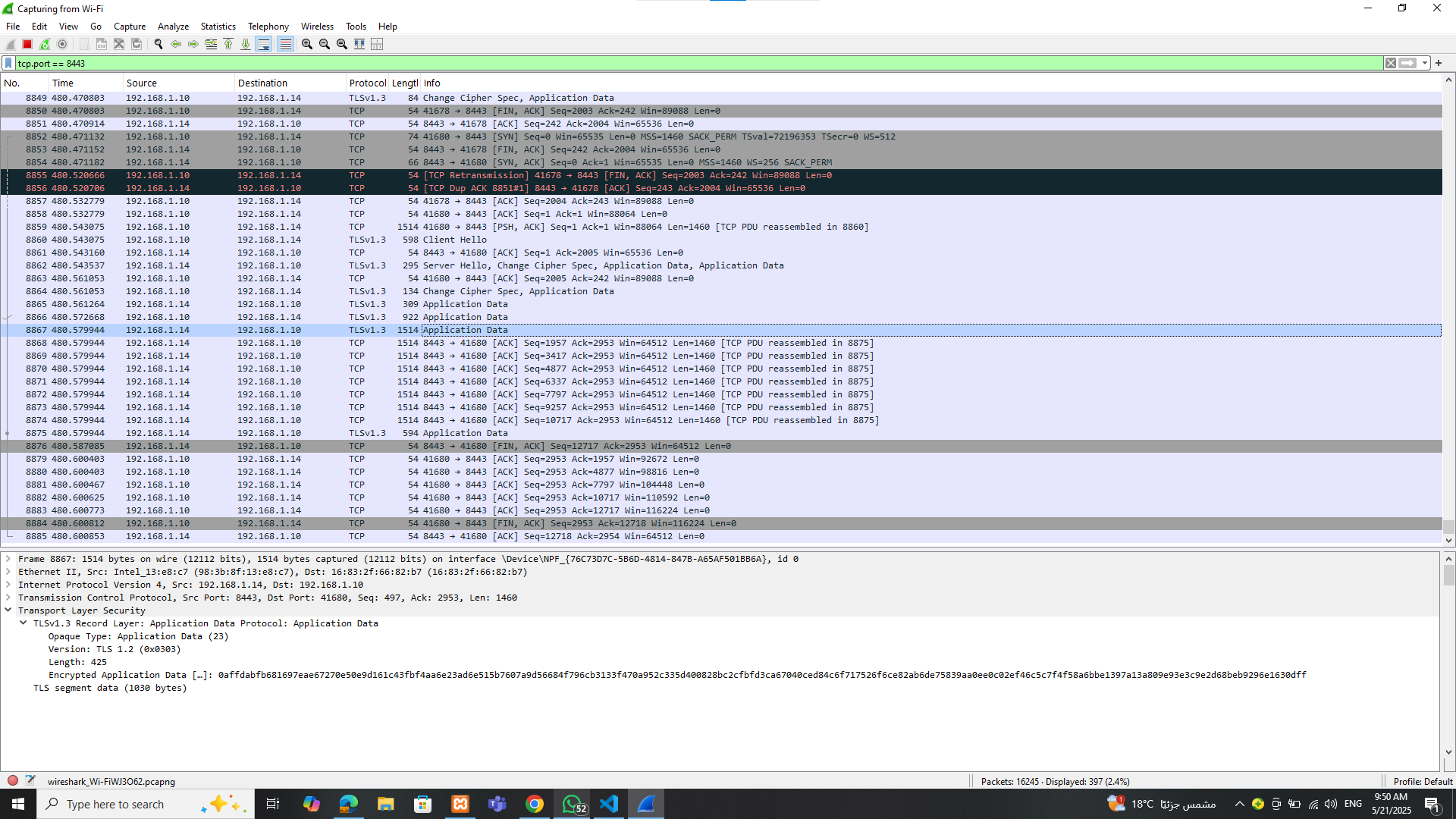
Wireshark displays the following:

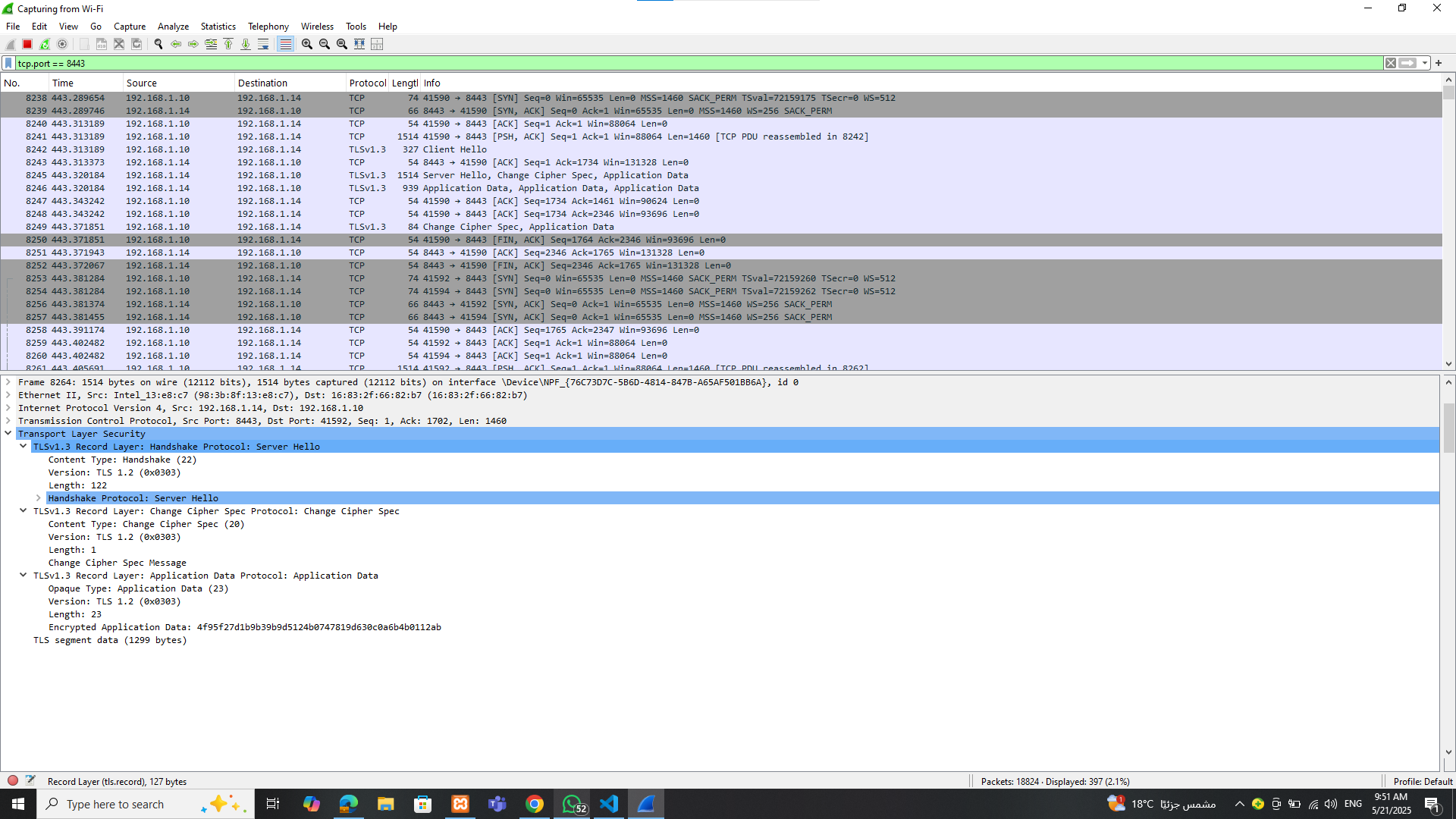
**Encrypted Handshake Message**

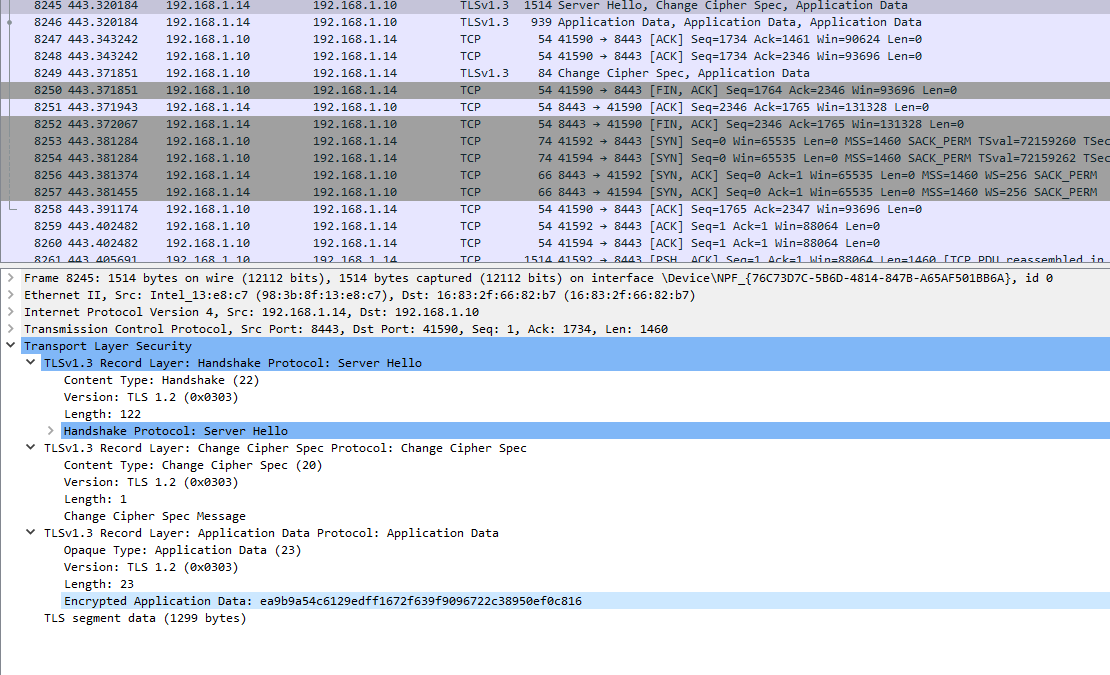
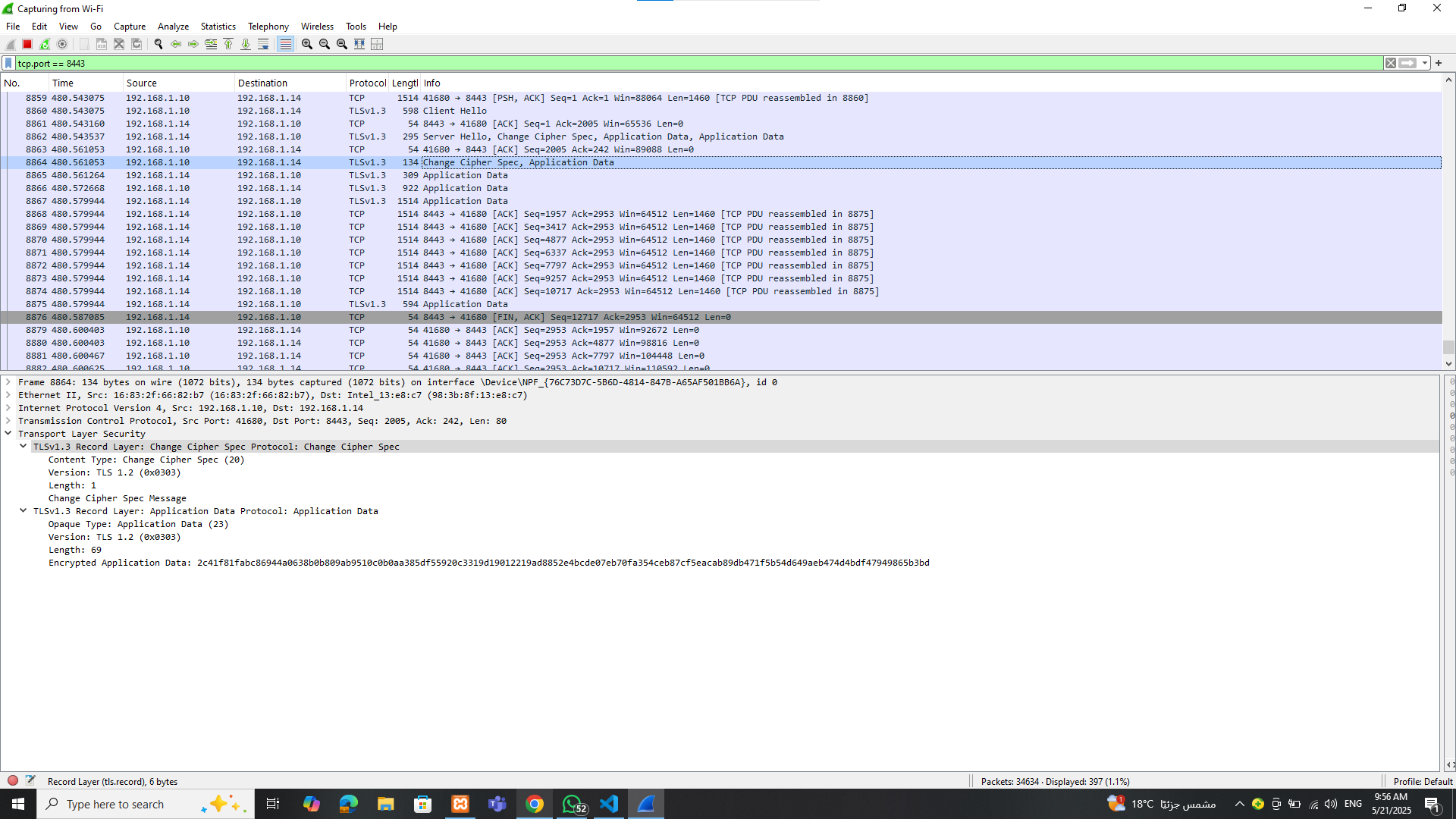
**Application Data**

**Explanation:**

* The login credentials and session data are fully encrypted.
* Even though the traffic is visible, its contents are unreadable.
* HTTPS ensures confidentiality and prevents MITM attacks







**Summary :**

This project simulated and analyzed the security behavior of a web login system under both HTTP and HTTPS protocols using Wireshark. The goal was to demonstrate how data travels over the network and how encryption protects it.

During the HTTP test, the login form was submitted without any encryption.  
 Wireshark successfully captured the full request, including the username and password, in plain text.  
 This proves that HTTP is highly vulnerable to Man-in-the-Middle (MITM) attacks. Anyone on the same network can intercept and read sensitive information.

In contrast, the HTTPS test showed a secure communication process.  
 The same login request was encrypted using TLS (Transport Layer Security), and Wireshark could no longer view the username or password.  
 Only encrypted messages and TLS handshake steps (such as “Client Hello” and “Encrypted Handshake Message”) were visible.  
 This confirms that HTTPS provides confidentiality, integrity, and protection against data interception.

Additionally, the login system implemented a Two-Factor Authentication (2FA) mechanism.  
 After entering correct credentials, the user was required to submit a 6-digit verification code.  
 This code is generated via a trusted method (authenticator app or email), and prevents unauthorized access even if the password is compromised.  
 The 2FA step was successfully tested and confirmed working, adding a strong layer of protection

**Conclusion:**

* HTTP allows full credential exposure — not safe for login forms.
* HTTPS encrypts all traffic — making data unreadable to attackers.
* 2FA adds another layer of defense by requiring a second verification step.

The experiment confirms that combining HTTPS with 2FA significantly improves user account security and protects against common network attacks.