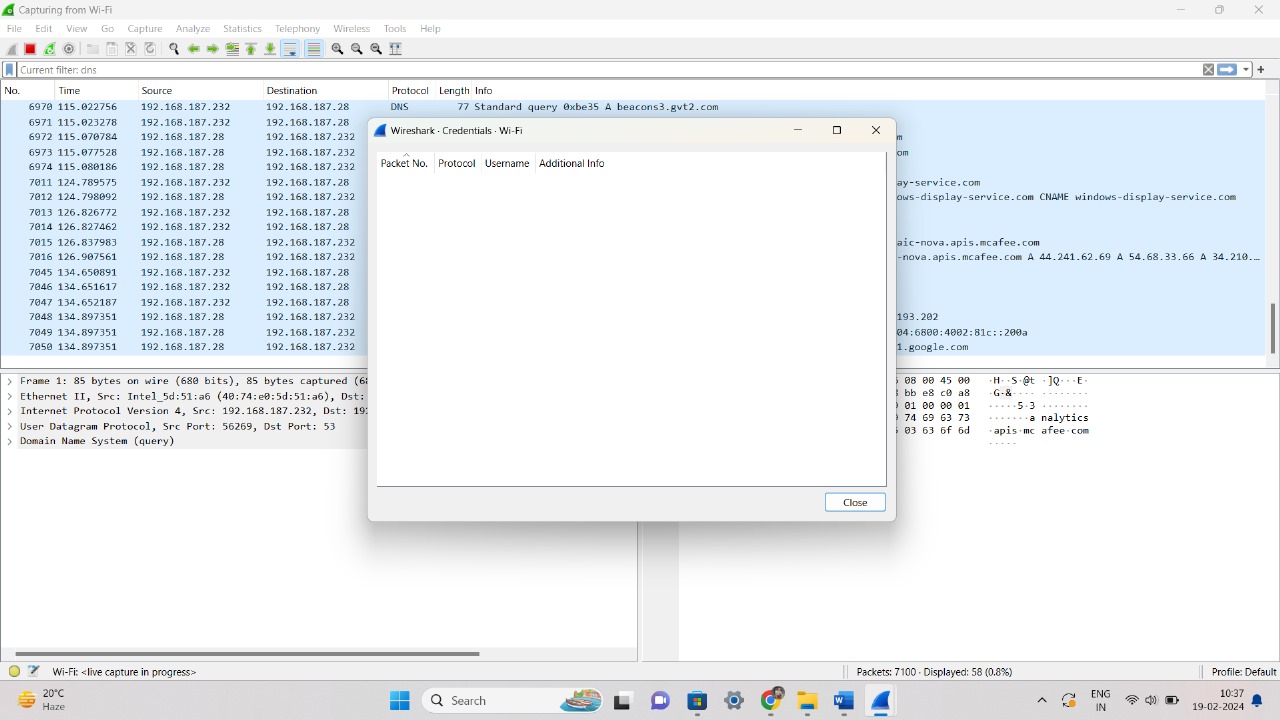
**Experiment-5**

**Topic given to me:** Credentials

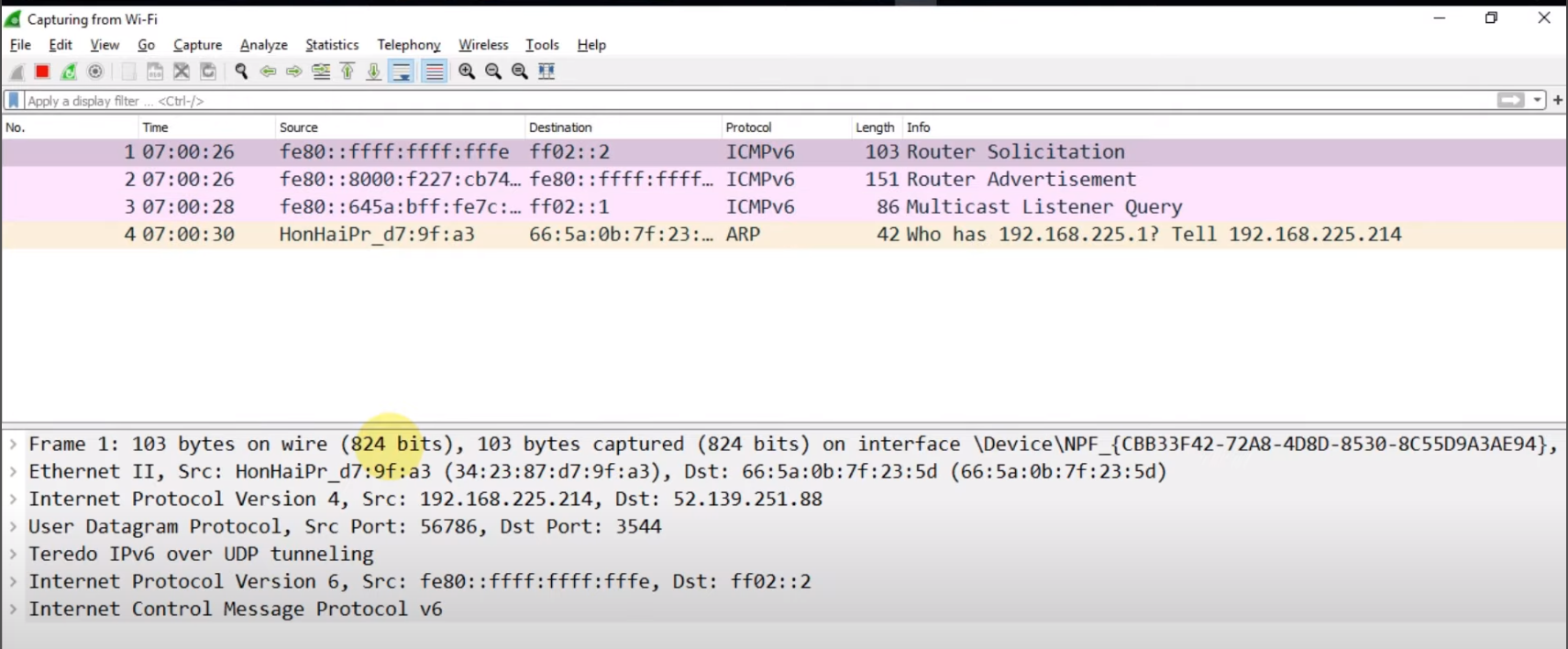
In Wireshark, credentials refer to the usernames and passwords transmitted over a network in plaintext or in an encrypted form. Capturing credentials is one of the significant use cases of Wireshark in security analysis and troubleshooting. Understanding the use of credentials in Wireshark can be broken down into several key points:

* **Password-based Authentication:** Many network protocols, such as HTTP, FTP, Telnet, SMTP, POP3, IMAP, etc., transmit credentials (username and password) in plaintext or encrypted form during authentication. Wireshark can capture and analyze these packets to reveal the credentials being transmitted.
* **Security Analysis:** Wireshark allows security professionals to capture network traffic and analyze it to identify any instances of insecure credential transmission. This helps in identifying potential security vulnerabilities, such as using weak or easily guessable passwords, or transmitting credentials over unencrypted channels.
* **Troubleshooting:** When users encounter issues with authentication on a network, Wireshark can be used to capture and analyze the network traffic to determine if there are any problems with the authentication process. For example, if a user is unable to log in to a web application, analyzing the HTTP traffic using Wireshark may reveal any errors or issues with the authentication process.
* **Auditing:** Wireshark can be used for auditing purposes to monitor and analyze network traffic for unauthorized access attempts or suspicious activity. By capturing and analyzing credential-related traffic, organizations can identify potential security breaches and take appropriate action to mitigate them.
* **Training and Education:** Wireshark is a valuable tool for training and education in the field of network security. By capturing and analyzing credential-related traffic in controlled environments, students and professionals can gain hands-on experience in identifying and understanding different types of credential-based attacks and vulnerabilities.

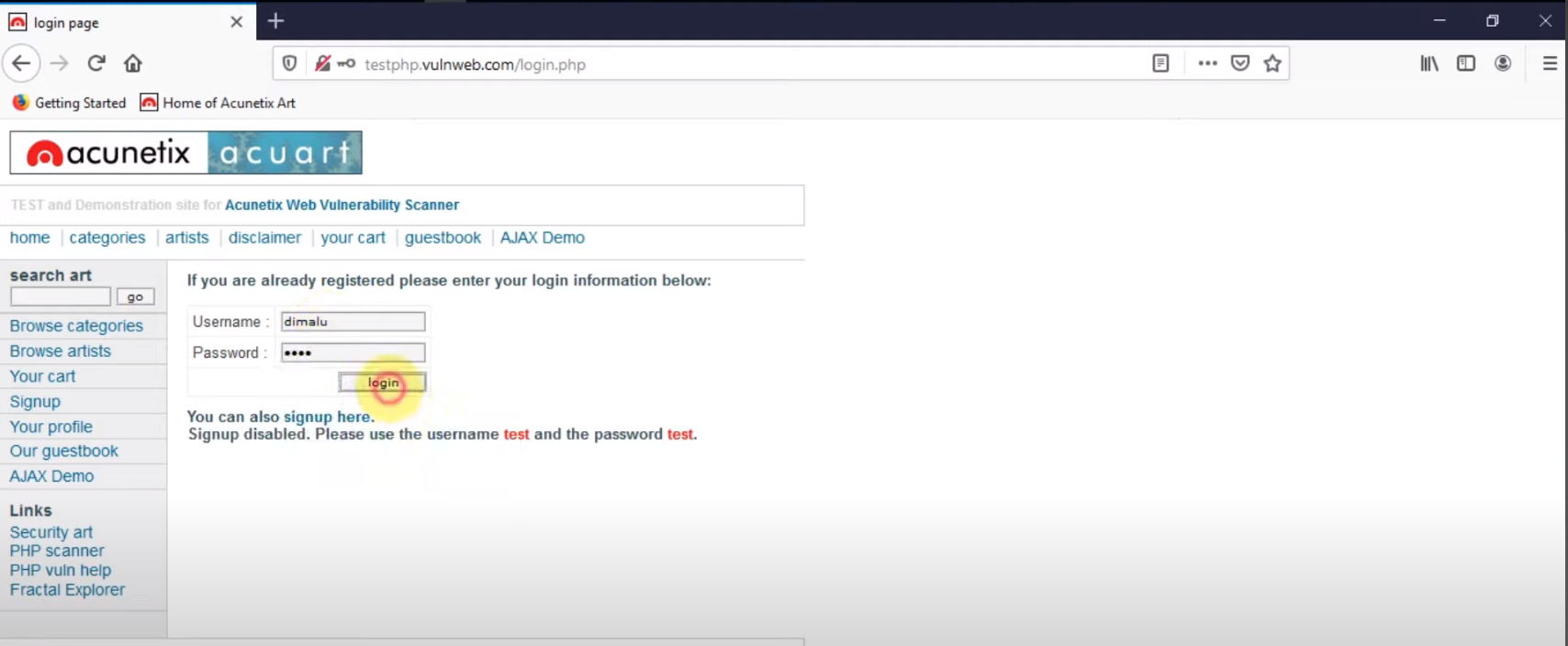


**Steps**

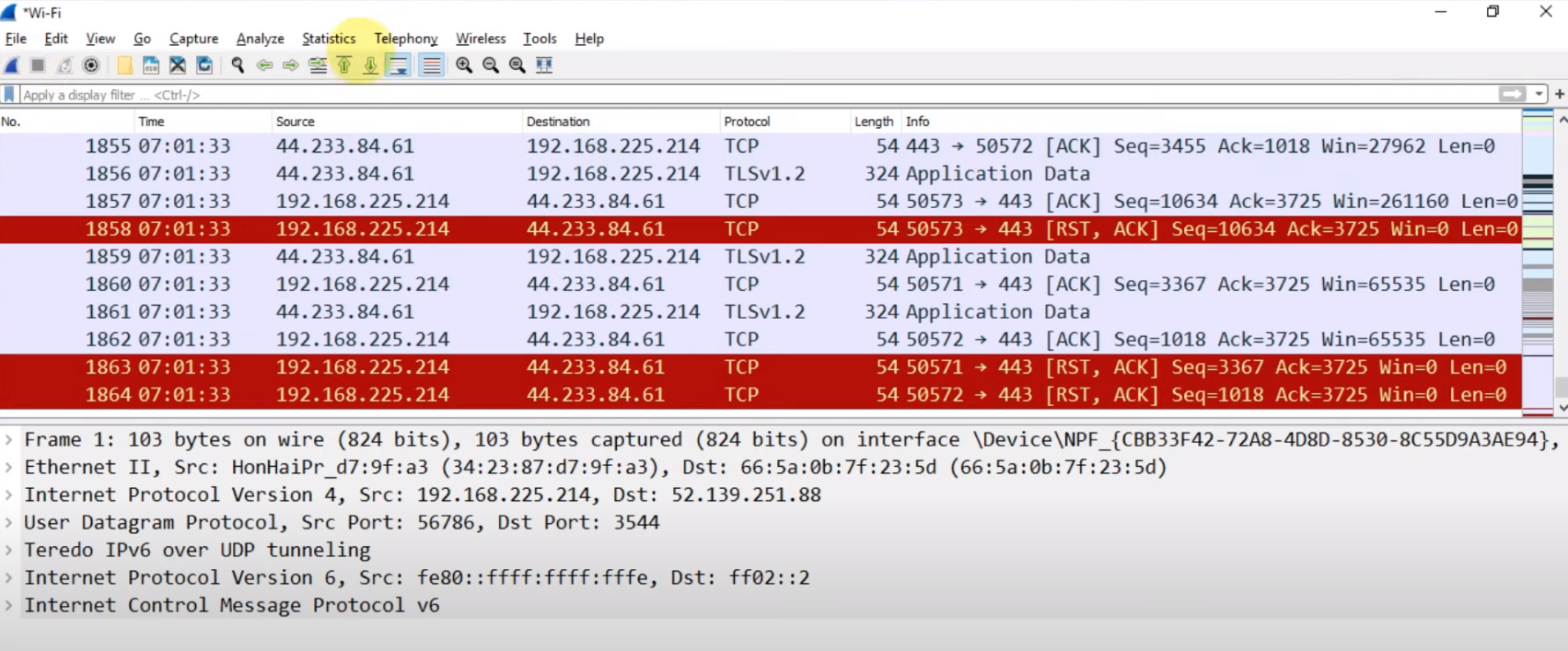
**Step-1:** Open wireshark and select the interface.



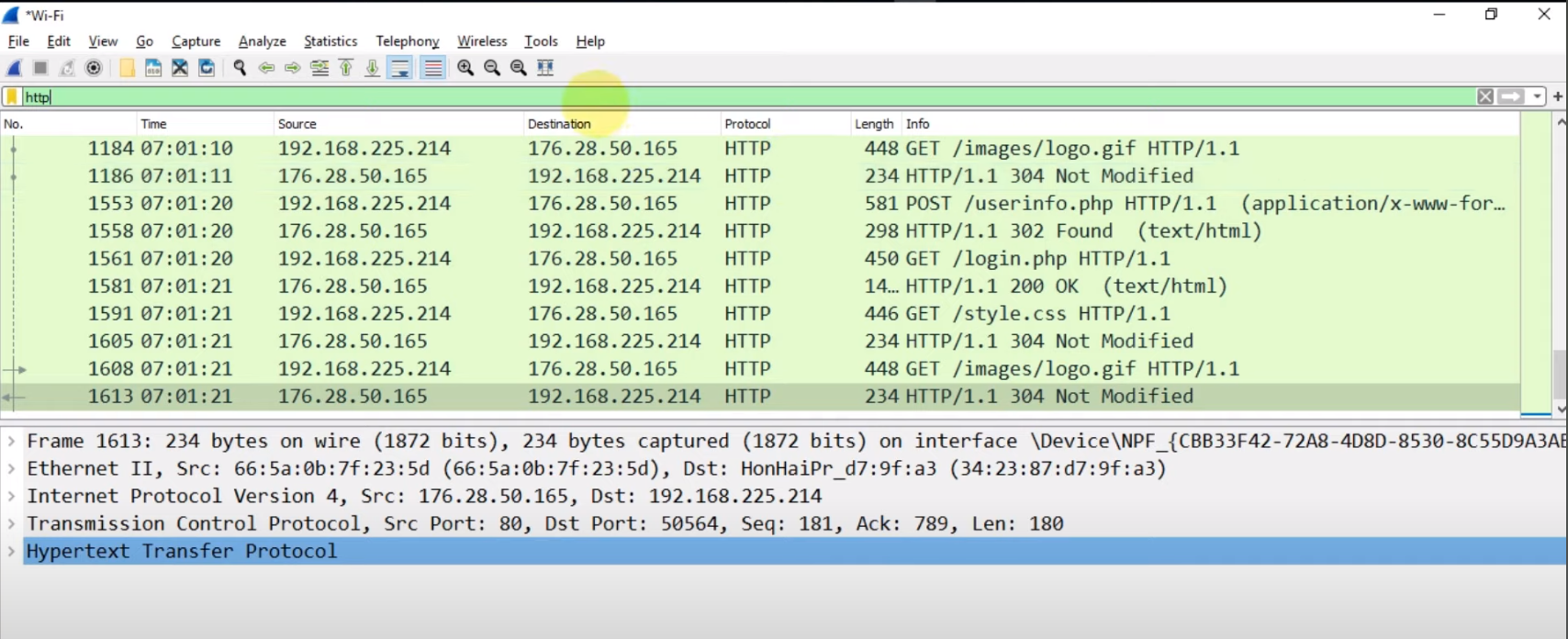
**Step-2:** Now go to browser and enter details(username and password) in a form.



**Step-3:** Now go back to the wireshark and here the wireshark would have captured several network packets.

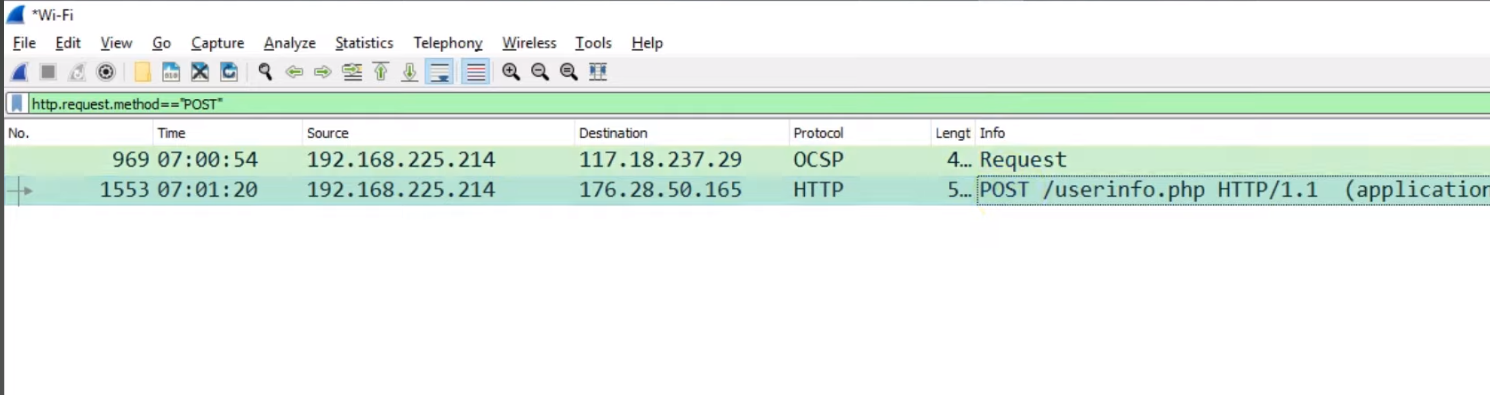
****

**Step-4:** Since we need the form data that we just submitted on login page, then we will select the http protocol.

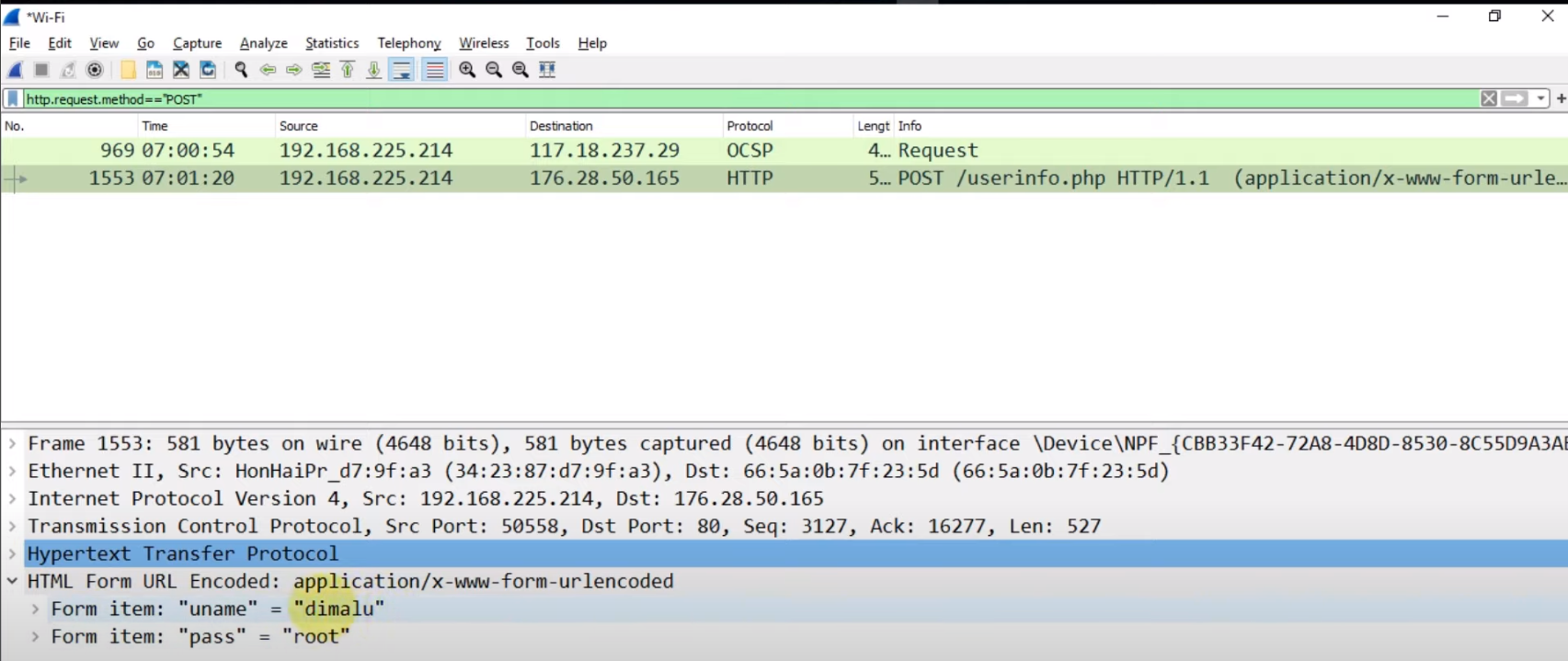
****

**Step-5:** Now the form details will be specifically captured using POST method. So we can use the filter:

http.request.method==”POST”

****

**Step-6:** The packet is displayed now we can select it and in the bottom tab, we can click on the HTML Form URL Encoded and the username and password will be displayed.

****

**Problem Statement:** Study and explore pen testing tools(wireshark/metaspoilt)

**Questions**

**1) What is Pen Testing, and How is it Useful to Ensure Security? What Are Different Pen Testing Tools?**

**Ans-** Penetration testing, commonly known as pen testing, is a proactive approach to evaluating the security of computer systems or networks by simulating real-world attacks. It involves identifying vulnerabilities in systems, networks, or applications and attempting to exploit them to gauge their potential impact. Pen testing is instrumental in ensuring security by uncovering weaknesses before malicious actors exploit them. Various pen testing tools serve different purposes:

* **Network Scanning Tools**: Examples include Nmap, which scans networks for open ports, services, and vulnerabilities.
* **Vulnerability Scanners:** Tools like Nessus identify and assess vulnerabilities in systems and networks.
* **Exploitation Frameworks:** Metasploit is a widely used framework for developing and executing exploits against remote targets.
* **Network Traffic Analyzers:** Wireshark is a powerful tool for capturing and analyzing network traffic, aiding in the detection of suspicious activities.

**2) What are the Primary Features of Wireshark?**

**Ans-** Wireshark is a comprehensive network protocol analyzer with the following primary features:

* **Real-time Packet Capture:** Captures and displays network packets in real-time.
* **Protocol Support:** Supports a vast array of protocols for detailed analysis of network traffic.
* **Flexible Capture Options:** Can capture packets from various network interfaces and save them for later analysis.
* **Robust Filtering Capabilities:** Provides rich filtering capabilities to isolate and focus on specific types of traffic.
* **Detailed Packet Inspection**: Allows detailed inspection of packet contents, including headers and payloads.
* **Live and Offline Analysis:** Supports both live capture and offline analysis of packet captures.

**3) How Do You Capture Network Traffic in Wireshark?**

**Ans-** Capturing network traffic in Wireshark involves selecting the desired network interface and starting a capture session. Users can:

1. Click on the appropriate interface in the main Wireshark window.
2. Initiate the capture session by clicking the "Start" button.
3. Alternatively, utilize the command-line interface for capture initiation.
4. Wireshark will begin capturing packets flowing through the selected interface in real-time.

**4) Explain the Purpose of Display Filters in Wireshark.**

**Ans-** Display filters in Wireshark serve to selectively display packets based on specific criteria, facilitating focused analysis. Key purposes include:

* **Noise Reduction:** Filter out irrelevant traffic to focus on pertinent packets.
* **Traffic Isolation:** Isolate specific types of traffic for analysis.
* **Customization:** Tailor display to specific protocols, IP addresses, ports, etc.
* **Enhanced Analysis:** Facilitate detailed examination of captured traffic.

**5) What is the Difference Between a Capture Filter and Display Filter in Wireshark?**

**Ans-** In Wireshark:

* **Capture Filter:** Applied before packet capture, determines which packets are captured.
* **Display Filter:** Applied after packet capture, determines which packets are displayed.

Capture Filters are used to limit captured traffic, while Display Filters selectively display captured packets.

**6) How Can You Analyze the Details of Captured Packets in Wireshark?**

**Ans-** Wireshark offers numerous tools for detailed packet analysis:

* **Packet Inspection:** Users can inspect headers, payloads, and other packet details.
* **Timing Analysis:** View packet timing information to understand network behavior.
* **Stream Analysis:** Follow TCP streams or analyze packet sequences for deeper insights.
* **Drill-down Capability:** Drill down into individual packets for specific details.

**7) What Security Considerations Should Be Taken Care of While Using Wireshark?**

**Ans-** When using Wireshark:

* **Authorization:** Ensure proper authorization before capturing network traffic.
* **Privacy Compliance:** Respect privacy laws and organizational policies regarding data interception.
* **Data Security:** Secure captured packet files to prevent unauthorized access.
* **Sensitive Information:** Exercise caution to avoid exposing sensitive information during packet capture and analysis.

**8) What Are npCap and WinPcap?**

**Ans-** WinPcap and npCap are packet capture libraries for Windows operating systems that provide low-level network access for applications like Wireshark. WinPcap was the original packet capture library for Windows but is no longer actively maintained. npCap is a successor to WinPcap, offering improved performance, compatibility with modern Windows versions, and ongoing development and support. Both libraries allow applications like Wireshark to capture and analyze network traffic on Windows systems.

WinPcap: Original library for low-level network access, now discontinued.

npCap: Successor to WinPcap, offering improved performance and compatibility.

Both libraries enable applications like Wireshark to capture and analyze network traffic on Windows systems.