Vulnerability Test Report: Capturing Passwords via HTTP

# 1. Introduction

This vulnerability assessment simulates a Man-in-the-Middle (MITM) attack to capture login credentials transmitted over unsecured HTTP connections. The goal is to demonstrate the risk of sending sensitive data over unencrypted channels.

# 2. Environment Setup

The test was conducted in a controlled lab environment using:

- Attacker Machine: Kali Linux  
- Victim Machine: Ubuntu  
- Network Type: Host-only network (VM Ware)

# 3. Tools Used

- Ettercap (GUI version)  
- Wireshark  
- Test HTTP login website (e.g., testphp.vulnweb.com)

# 4. Procedure

Step-by-step actions taken:

1. Enabled IP forwarding on the attacker (Kali) machine:  
 echo 1 | sudo tee /proc/sys/net/ipv4/ip\_forward

2. Launched Ettercap in GUI mode:  
 sudo ettercap -G

3. Selected the appropriate network interface (e.g., eth0 or ens33).

4. Scanned for live hosts and identified the victim and gateway.

5. Set Target 1 (victim IP) and Target 2 (gateway IP).

6. Enabled ARP Poisoning via Mitm → ARP Poisoning.

7. Started packet sniffing: Start → Start Sniffing.

8. Monitored the victim accessing an HTTP login form via **“http://testphp.vulnweb.com”**

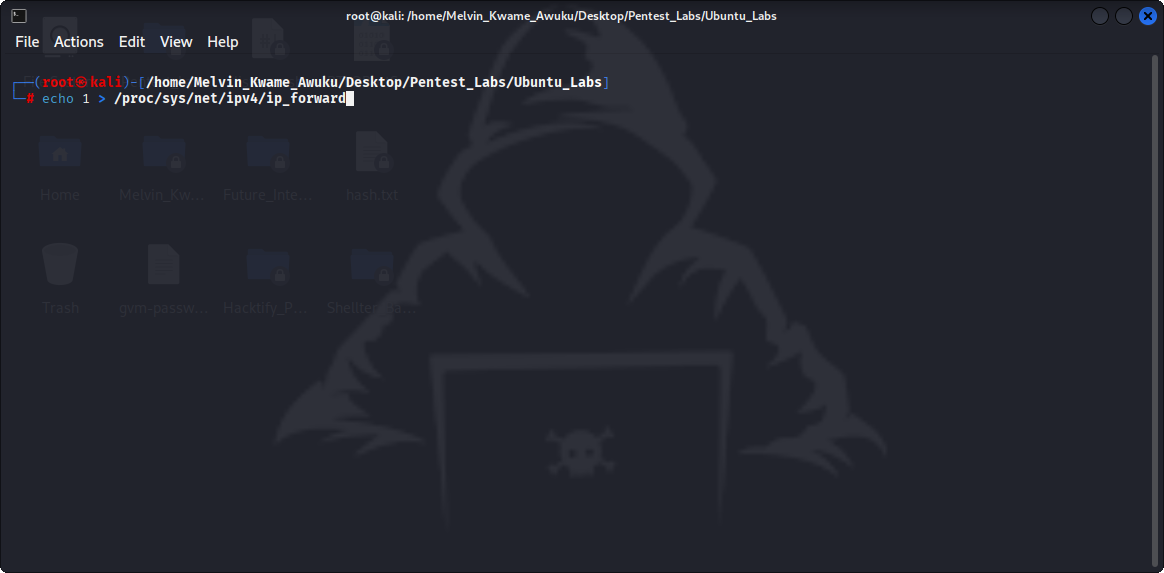
9. Observed credentials captured in Ettercap logs.

# 5. Observations

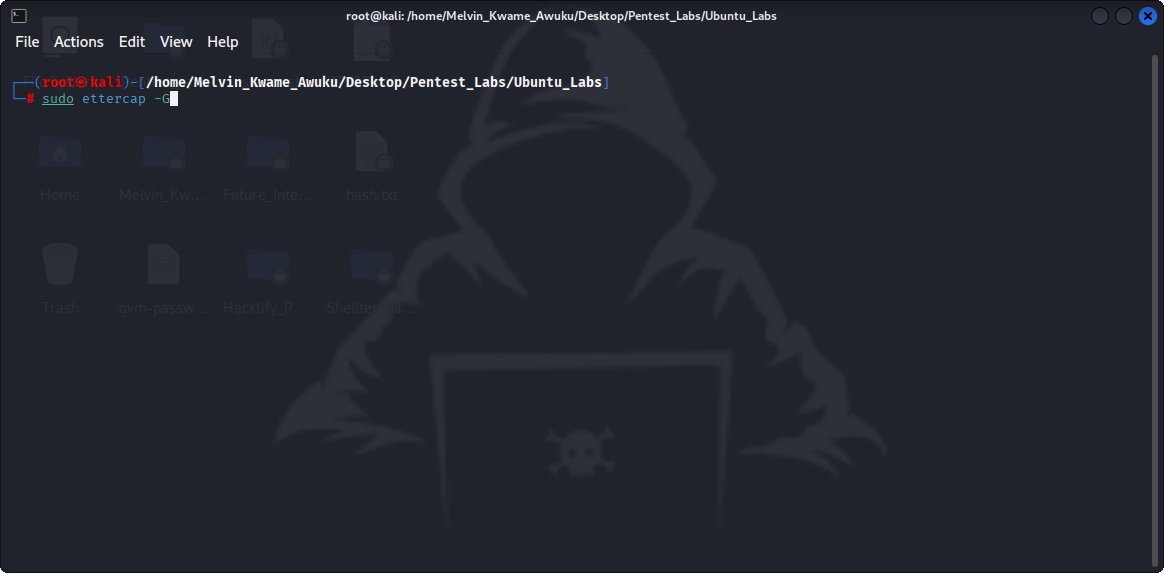
Credentials such as usernames and passwords transmitted via HTTP were successfully intercepted using Ettercap. Although Wireshark was used to attempt inspection, the POST request did not appear, indicating either HTTPS encryption or alternative form submission method.

# 6. Screenshots

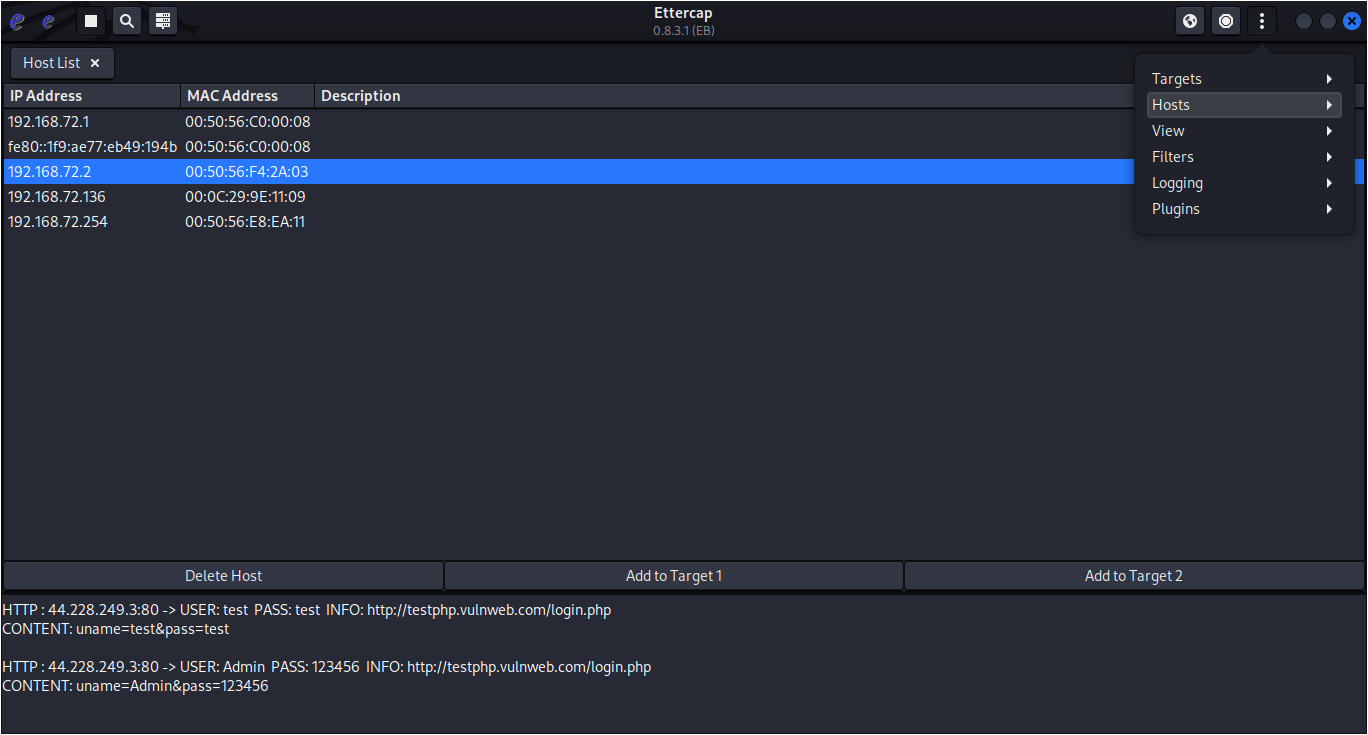
The following screenshots were taken to document the procedure:



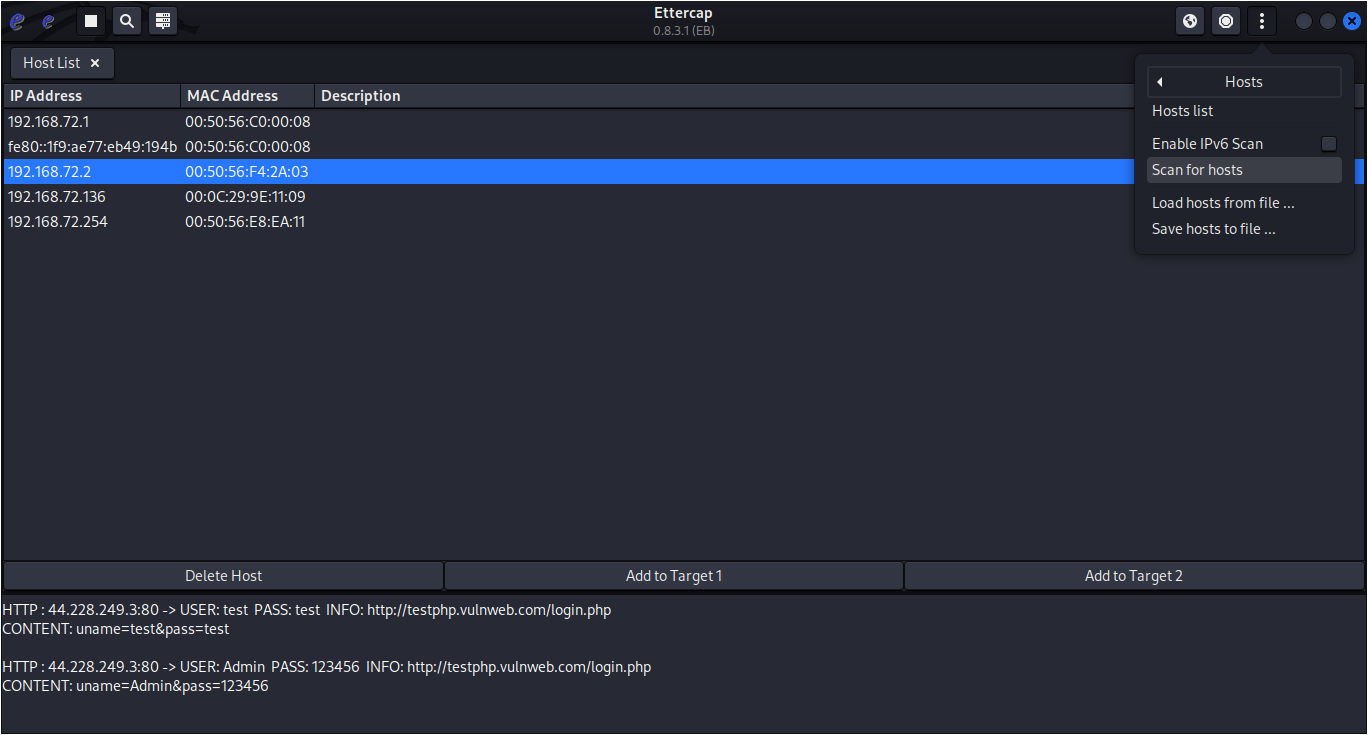
Image\_1.png



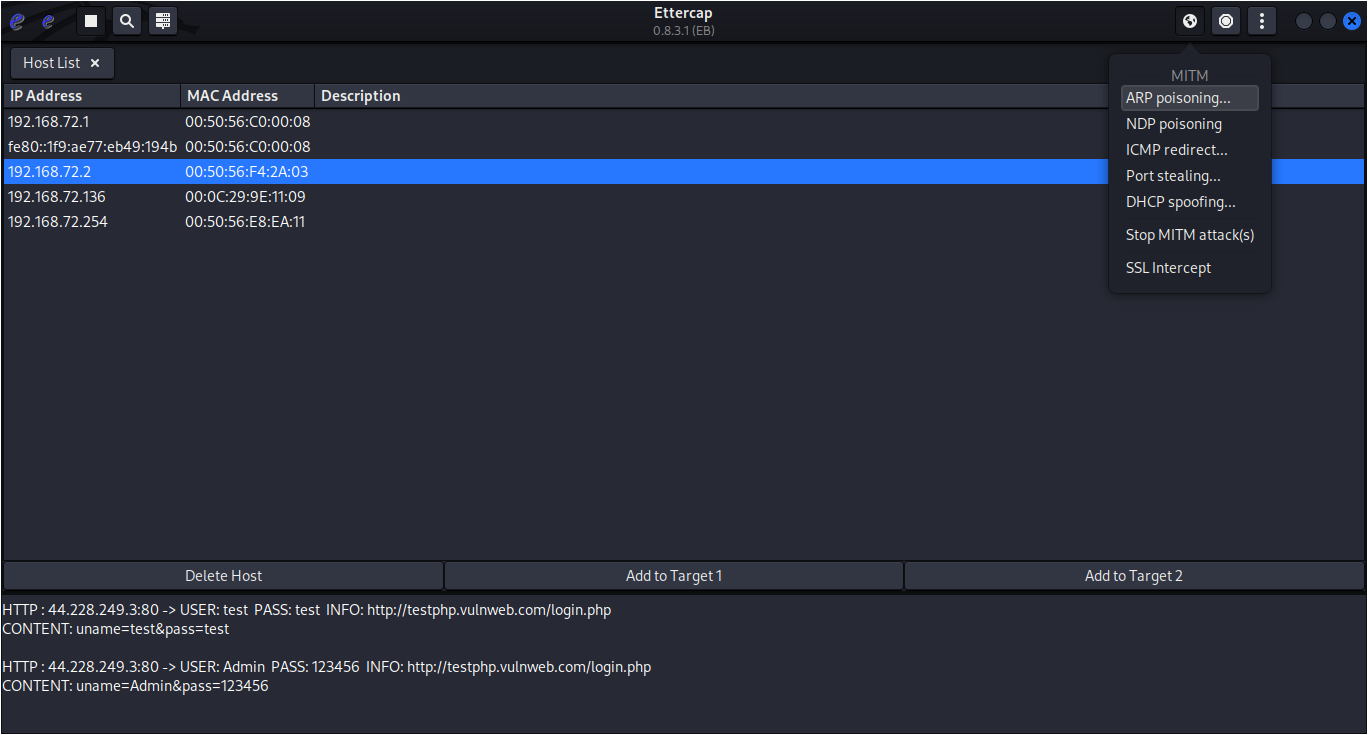
Image\_2.png



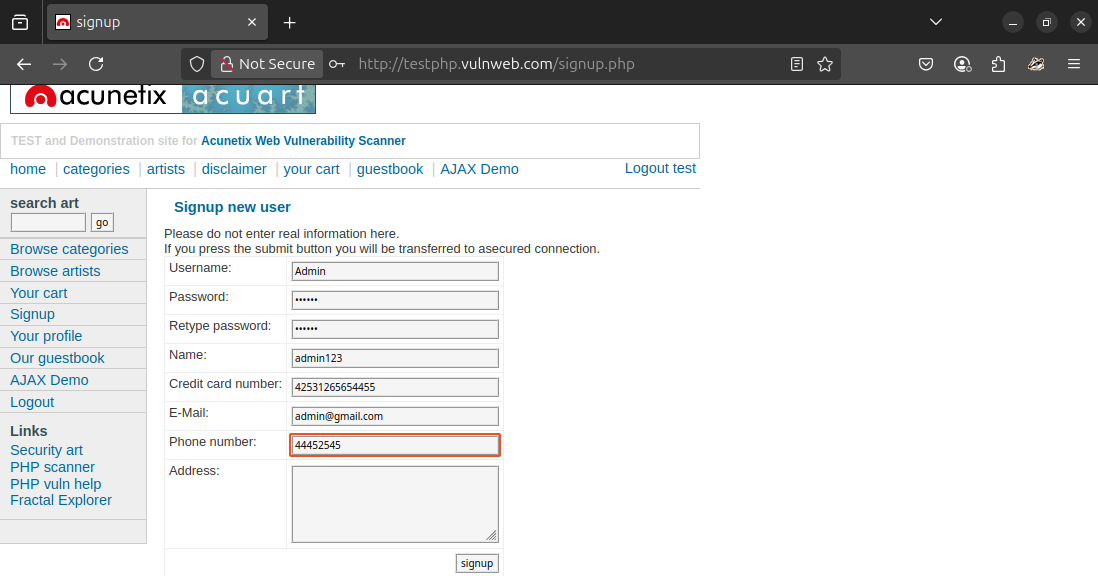
Image\_3.png



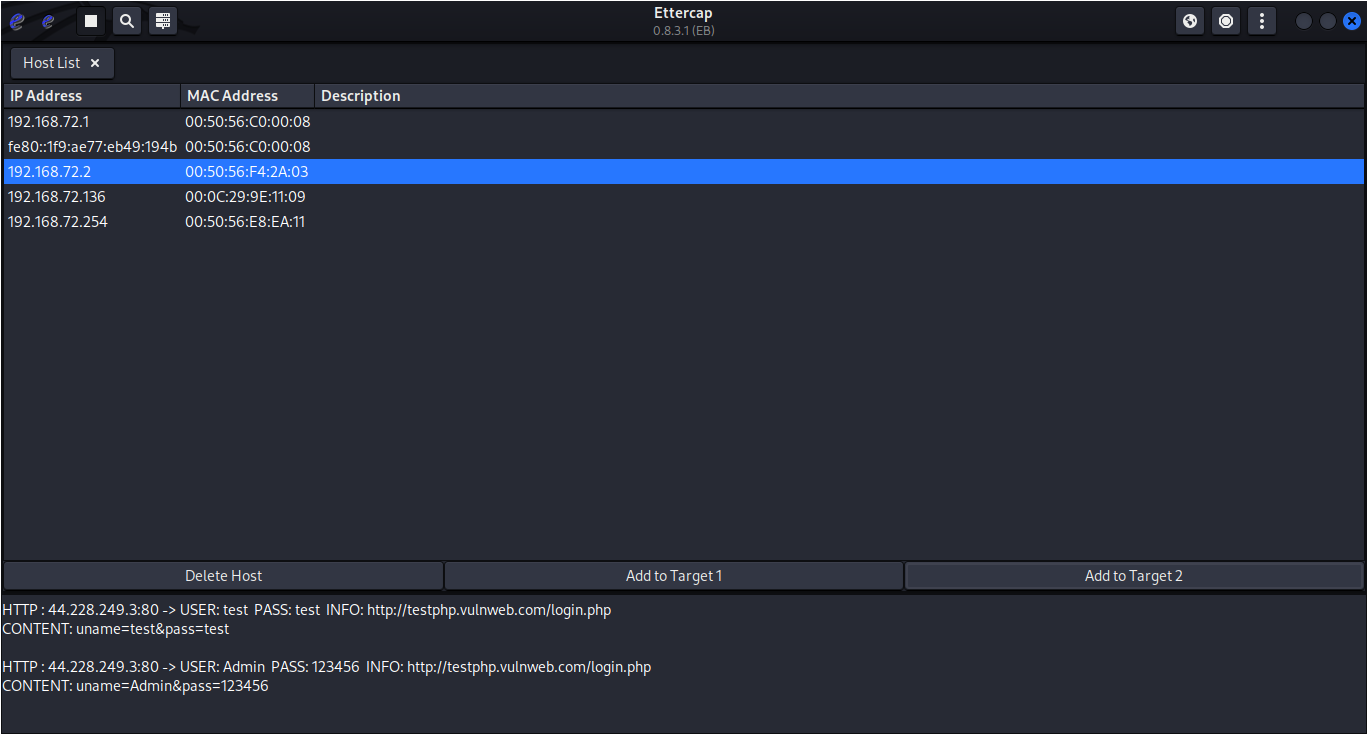
Image\_4.png



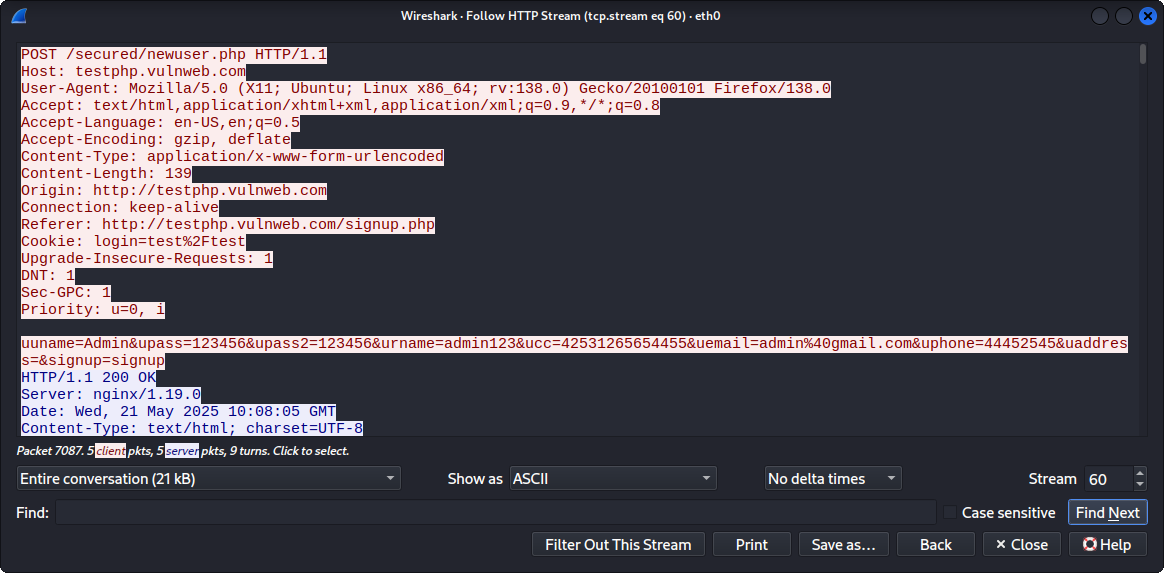
Image\_5.png



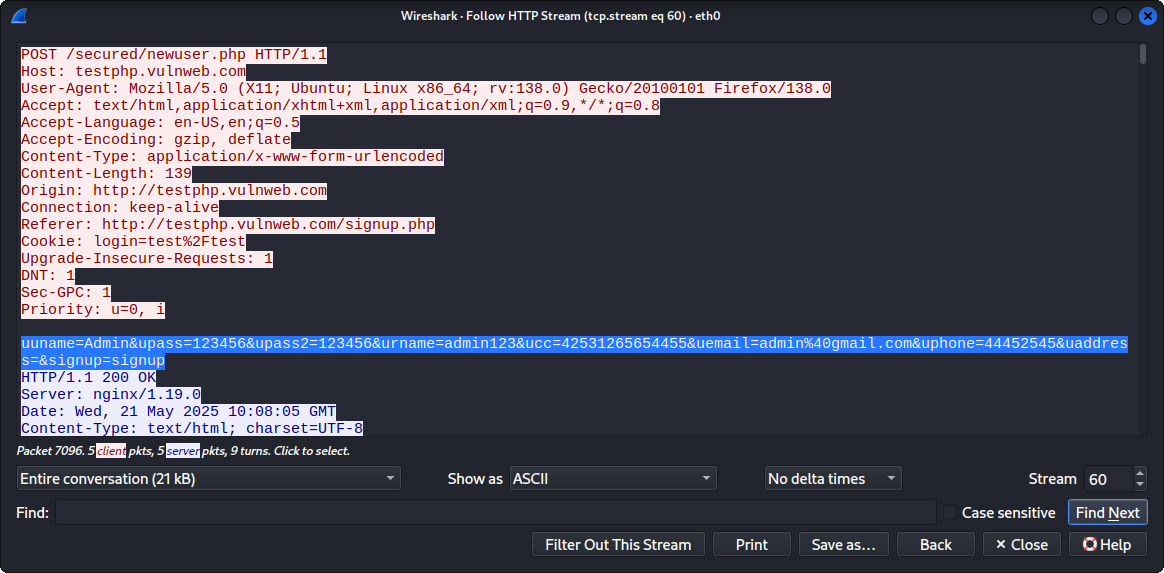
Image\_6.png



Image\_7.png



Image\_8.png



Image\_9.png

# 7. Conclusion

This test demonstrates how easily credentials can be stolen when transmitted over HTTP. It reinforces the importance of always using HTTPS for secure communications.

# 8. Recommendations

- Use HTTPS across all websites, especially on login pages.  
- Implement HSTS (HTTP Strict Transport Security).  
- Educate users on verifying site security before entering credentials.  
- Monitor networks for ARP poisoning attempts.