Wireshark Traffic Capture Penetration Testing Report

# Executive Summary

During a traffic capture analysis using Wireshark, an HTTP-based authentication transaction was intercepted. The packet capture revealed serious security vulnerabilities, notably the transmission of credentials in cleartext over an unencrypted HTTP channel. Such issues can lead to credential theft and compromise of user accounts if an attacker is on the same network.

# 1. Scope

- Tool Used: Wireshark  
- Objective: Analyze traffic from a login page to identify potential security flaws in transmission.  
- Target: Web application transmitting credentials  
- Test Method: Passive packet capture and protocol analysis

# 2. Methodology

Captured HTTP traffic during login activity using Wireshark.  
Analyzed packets in the TCP stream to inspect transmitted data.  
Reassembled and reviewed HTTP requests to find sensitive information.

# 3. Key Findings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Vulnerability | Description | Severity | Evidence |
| 1 | Cleartext Credential Transmission | Username and password are sent via HTTP, which lacks encryption. This allows any attacker monitoring the network to intercept user credentials. | High | Screenshot showing HTTP POST request with visible credentials |
| 2 | No TLS/SSL Protection | The communication occurs over HTTP instead of HTTPS, indicating absence of transport layer encryption. | High | TCP stream shows full unencrypted headers and body |
| 3 | Session Hijacking Risk | Without encryption, session cookies (if present) could be hijacked by intercepting the HTTP traffic. | Medium | Risk inferred due to lack of HTTPS (not shown in screenshots but assumed) |

# 4. Recommendations

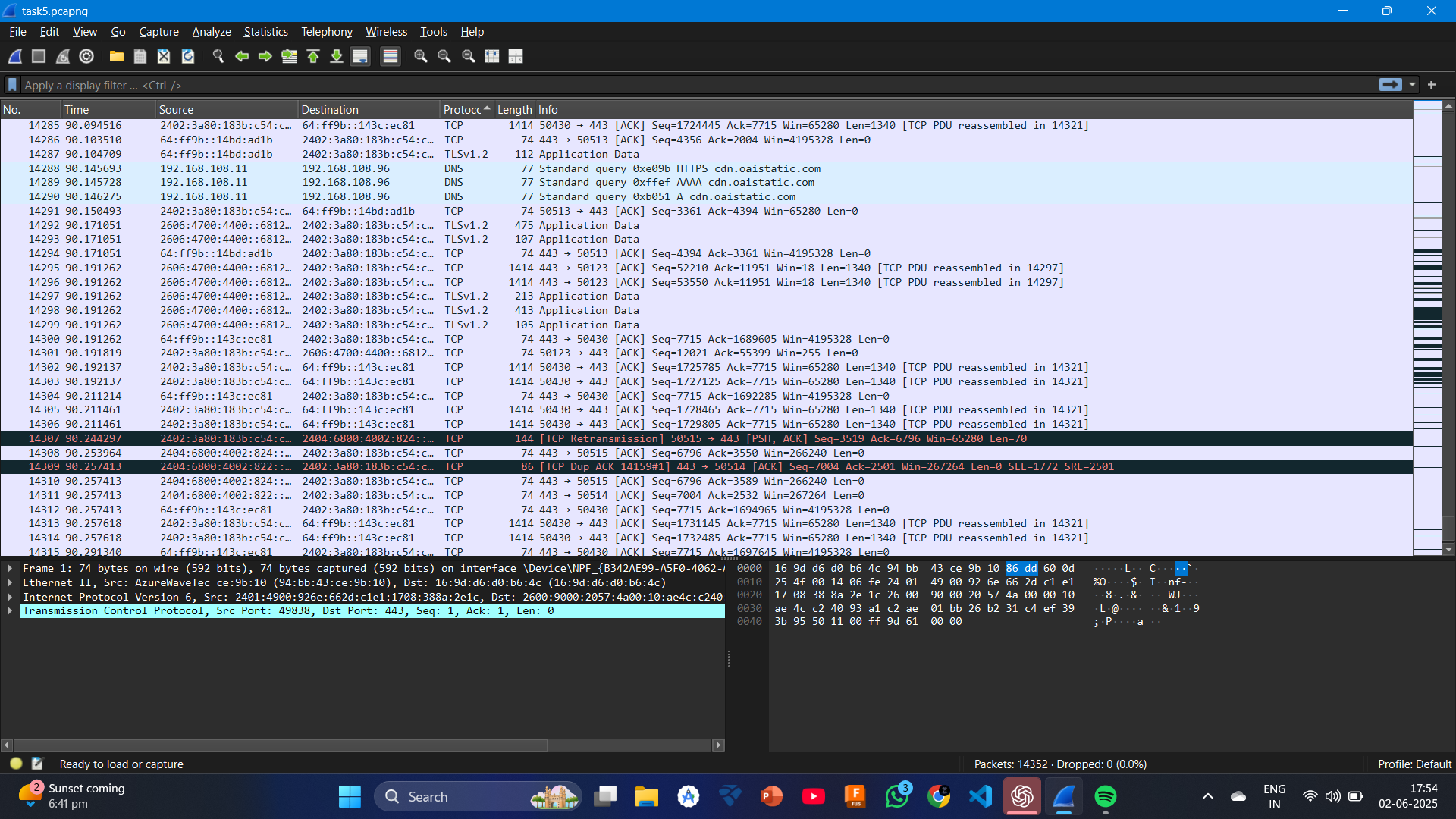
- Implement HTTPS (TLS) on all pages, especially login and sensitive actions.  
- Use HSTS (HTTP Strict Transport Security) to enforce secure connections.  
- Avoid transmitting credentials via GET or insecure POST methods.  
- Ensure secure cookie attributes (HttpOnly, Secure) are set.  
- Educate users about using secure networks (avoid public Wi-Fi without VPN).

# 5. Conclusion

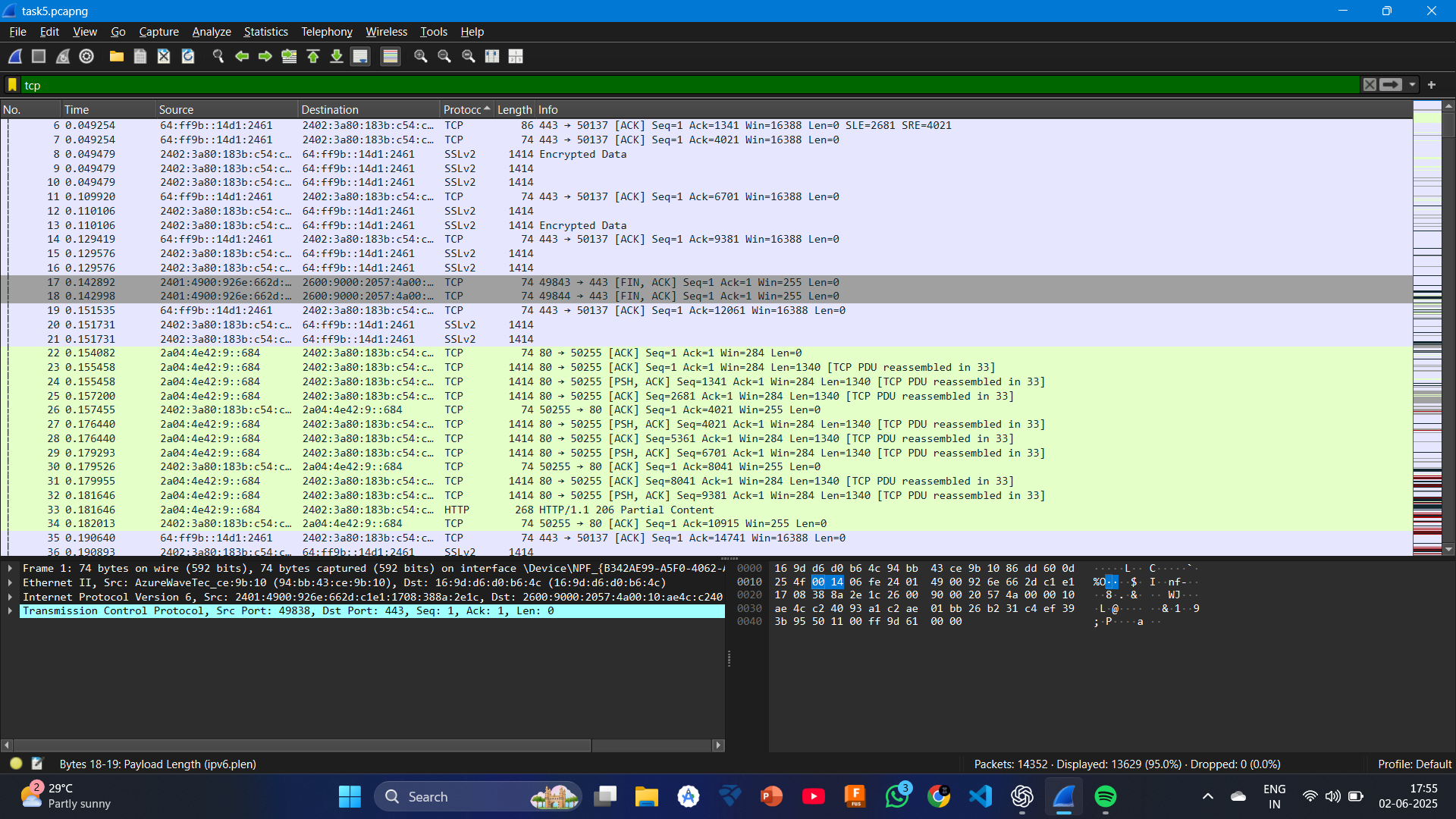
The captured traffic clearly demonstrates a critical flaw in the current web application security design. Using HTTP for authentication exposes user credentials and session data to attackers with basic packet-sniffing capabilities. Immediate action is necessary to switch to secure communication protocols (HTTPS) and enforce end-to-end encryption.

# Appendix: Packet Capture Screenshots

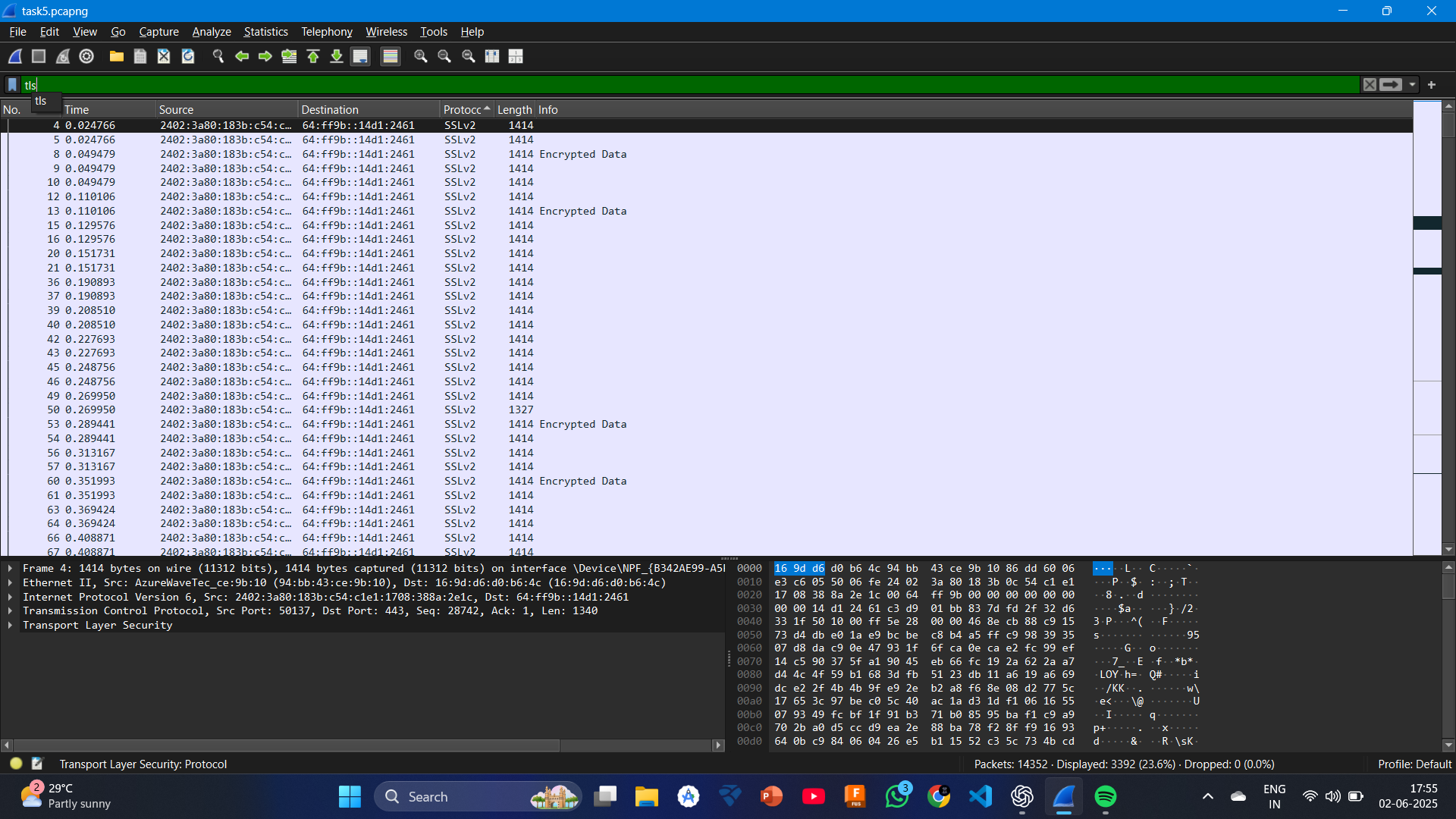
Below are the screenshots from Wireshark that support the findings listed in the report.



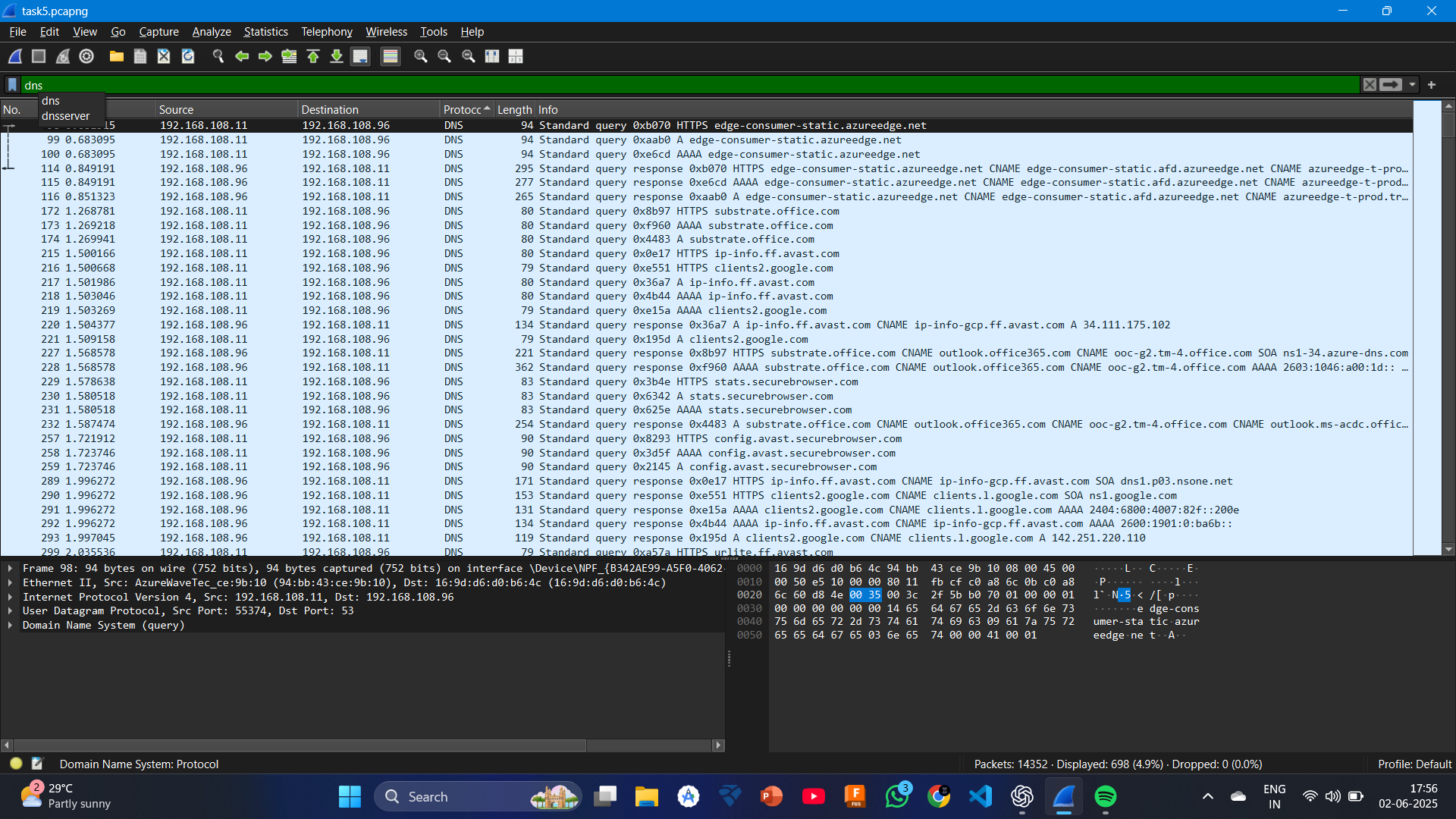
Screenshot 1: HTTP Login Request Overview



Screenshot 2: Packet Details with Cleartext Data



Screenshot 3: TCP Stream View of Login Request



Screenshot 4: Full HTTP Header and Body