

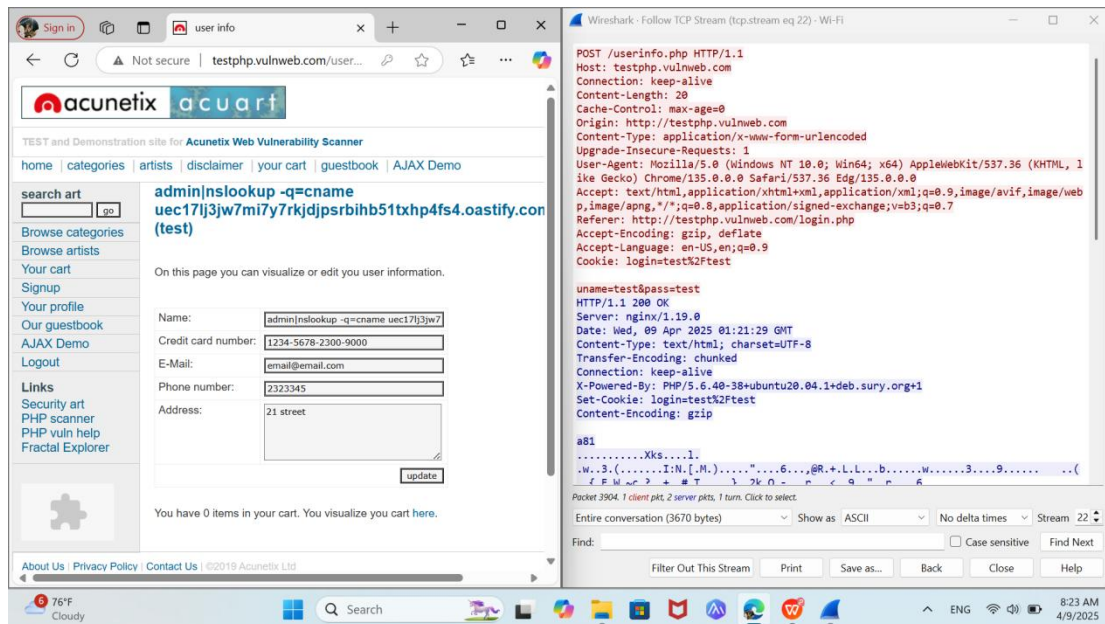
Nguyễn Thị Phương Anh-22174600085
Lê Thị Phương Linh-22174600057

Bước 1:

The screenshot displays a Wireshark network capture of a POST request to `/userinfo.php` on the host `testphp.vulnweb.com`. The packet list on the left shows the selected packet (No. 451) at time 17.389275. The packet details pane on the right shows the request structure, including headers like `Cache-Control: max-age=0`, `Origin: http://testphp.vulnweb.com`, `Content-Type: application/x-www-form-urlencoded`, `User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/537.36`, and `Referer: http://testphp.vulnweb.com/login.php`. The request body is `uname=test&pass=test`. The packet bytes pane shows the raw data in hexadecimal and ASCII.

Below the Wireshark capture, the web application interface is shown. The browser address bar indicates the URL `testphp.vulnweb.com/userinfo.php`. The page title is "Acunetix acuart". The page content includes a search bar with the query `admin[nslookup -q=cname uec17lj3jw7mi7y7rkjdjpsrbihb51txhp4fs4.oastify.com (test)]`. The page also displays a form for user information, including fields for Name, Credit card number, E-Mail, Phone number, and Address, with an "update" button.

The bottom of the screenshot shows the Windows taskbar with the system clock at 8:44 AM on 4/9/2025.



Code:

```
import pyshark

def parse_layers(file_path):
    cap = pyshark.FileCapture(file_path, display_filter='eth || ip || ipv6')

    print(" Phân tích các gói tin tầng 2 và tầng 3 trong file .pcapng:\n")

    for i, pkt in enumerate(cap):
        if i >= 10:
            break

        print(f" Gói tin {i + 1}:")

        # Tầng 2: Ethernet
        if 'eth' in pkt:
            eth = pkt.eth
            print(" Layer 2 - Ethernet:")
            print(f" MAC nguồn: {eth.src}")
            print(f" MAC đích: {eth.dst}")
            print(f" Ether Type: {eth.type}")

        # Tầng 3: IPv4
        if 'ip' in pkt:
            ip = pkt.ip
            print(" Layer 3 - IPv4:")
            print(f" IP nguồn: {ip.src}")
            print(f" IP đích: {ip.dst}")
            print(f" Giao thức: {ip.proto}")

        # Tầng 3: IPv6
        elif 'ipv6' in pkt:
            ipv6 = pkt.ipv6
            print(" Layer 3 - IPv6:")
            print(f" IP nguồn: {ipv6.src}")
            print(f" IP đích: {ipv6.dst}")
            print(f" Next Header: {ipv6.nxt}")

        print("-" * 40)

    cap.close()

# Gọi hàm với đường dẫn file (bạn cần viết đúng đường dẫn nếu dùng trên máy mình)
parse_layers("tcp(1).pcapng")
```

Kết quả:

Phân tích các gói tin tầng 2 và tầng 3 trong file .pcapng:

Nhận xét:

Gói tin HTTP

Gói HTTP là một gói tin hoạt động ở tầng 7 - Application Layer trong mô hình OSI.

Giao thức sử dụng: HTTP (HyperText Transfer Protocol)

Tầng vận chuyển: TCP (thường là cổng 80)

Đặc điểm:

Dữ liệu truyền là văn bản thuần túy, có thể đọc được bằng mắt thường (GET, POST, Header, Body...).

Khi bắt gói trong Wireshark, bạn có thể xem toàn bộ nội dung của gói HTTP như tiêu đề (header), nội dung (body), cookie, v.v.

HTTP: Dễ đọc, không bảo mật.

HTTPS: Bảo mật, mã hóa toàn bộ nội dung.

Gói HTTPS cũng là giao thức tầng 7, nhưng có đặc điểm:

Giao thức: HTTPS = HTTP + TLS/SSL (bảo mật)

Tầng vận chuyển: TCP (thường là cổng 443)

Đặc điểm:

Dữ liệu được mã hóa, không đọc được bằng mắt thường.

Bạn không thể thấy nội dung HTTP bên trong nếu không có khóa giải mã.

Chỉ có thể thấy quá trình bắt tay TLS (TLS handshake), chứng chỉ, v.v.