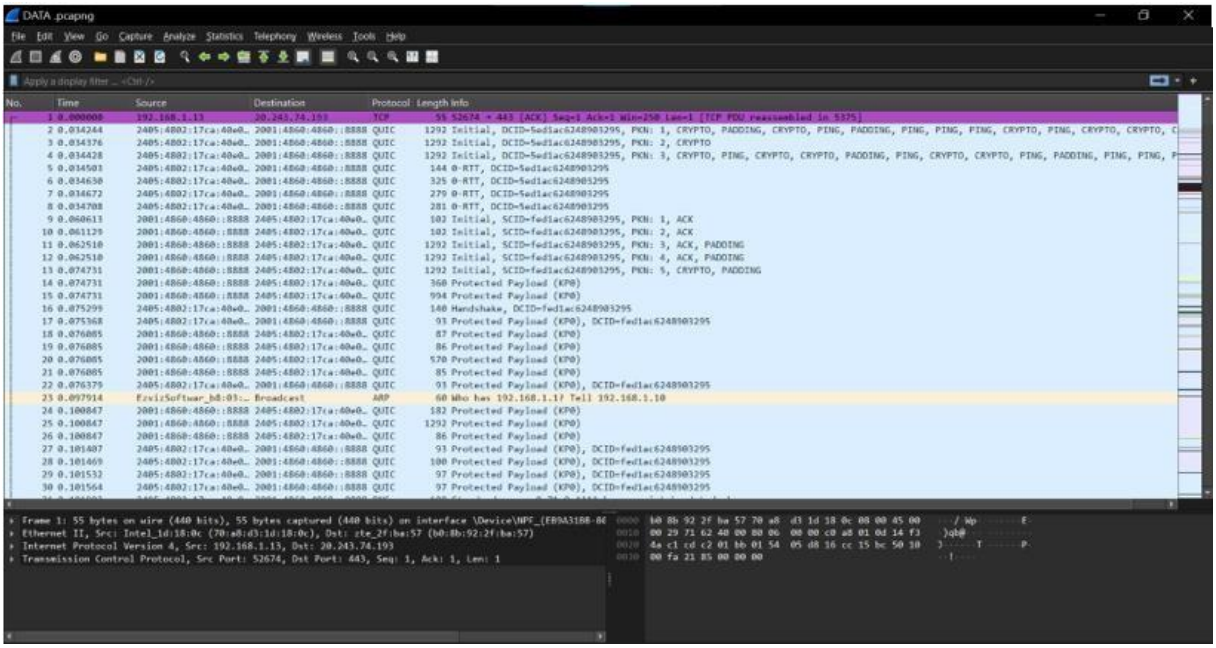
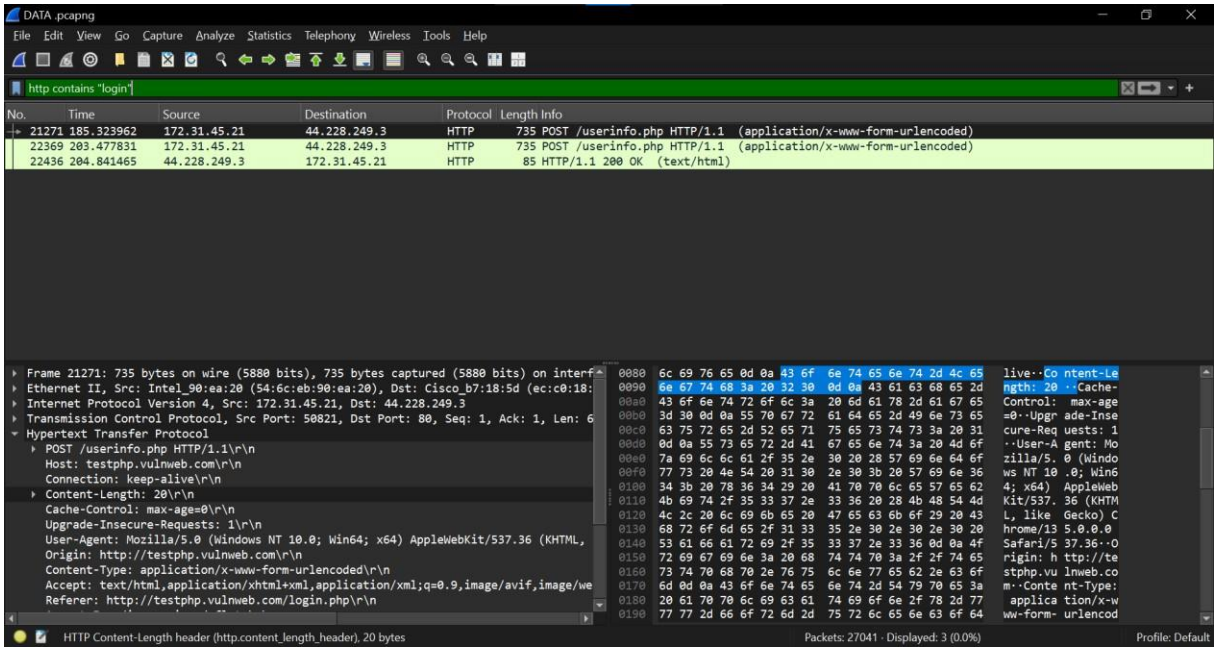
BÀI THỰC HÀNH 4

Họ và tên	Mã sinh viên
Lưu Nhật Nam	22174600109
Hà Quang Vinh	22174600065

Bước 1: Chọn card mạng Wifi



Bước 2: Lọc giao thức HTTP



Bước 3:

- Lưu file thành DATA.pcapng

Bước 4: Mở và trực quan hóa gói tin HTTP đã lưu

Mở file .pcapng đã lưu:

Vào menu File → Open, chọn file .pcap vừa lưu và nhấn Open.

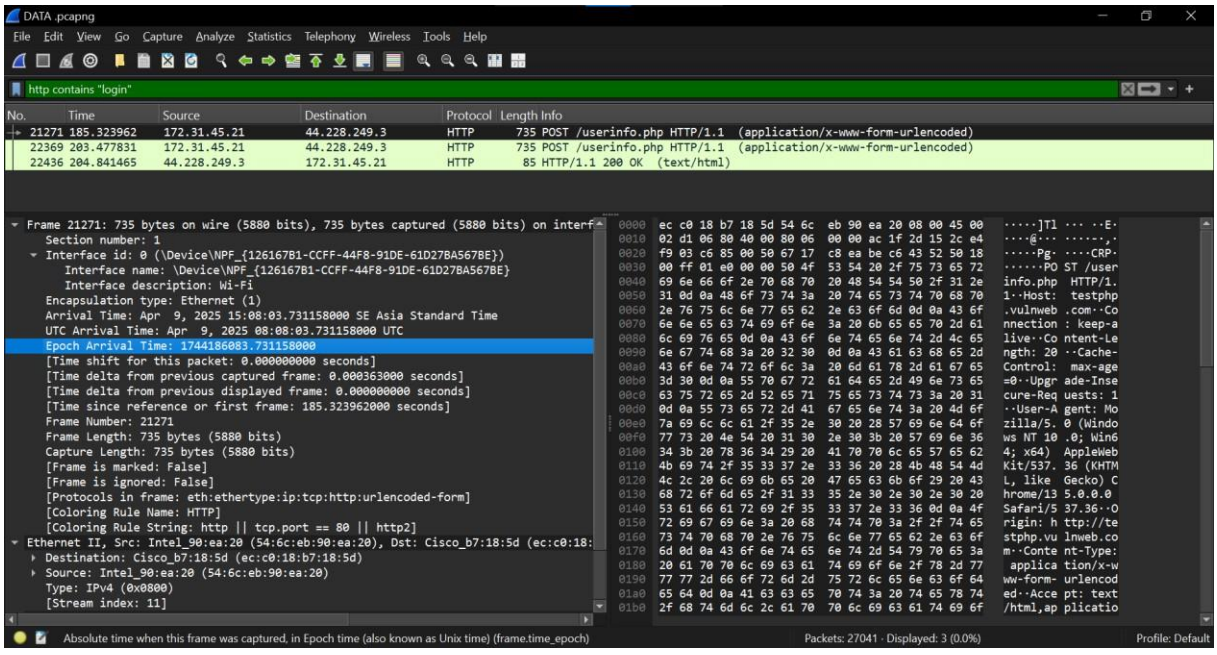
Trực quan hóa và phân tích các trường của gói tin:

Gói cần phân tích là gói số 21271 – POST / userinfo.php, đã được bạn bắt thành công.

Frame: Thông tin chung như thời gian, độ dài gói tin.

Thông tin chung: 21271 (số thứ tự trong file .pcapng)

Thời gian bắt được 185.323962 giây kể từ khi bắt đầu bắt gói
Độ dài gói trên dây (Wire length)
735 bytes (5880 bits) – tổng kích thước khi truyền qua mạng
Độ dài đã bắt được (Captured length)
735 bytes – không bị mất dữ liệu trong quá trình bắt gói



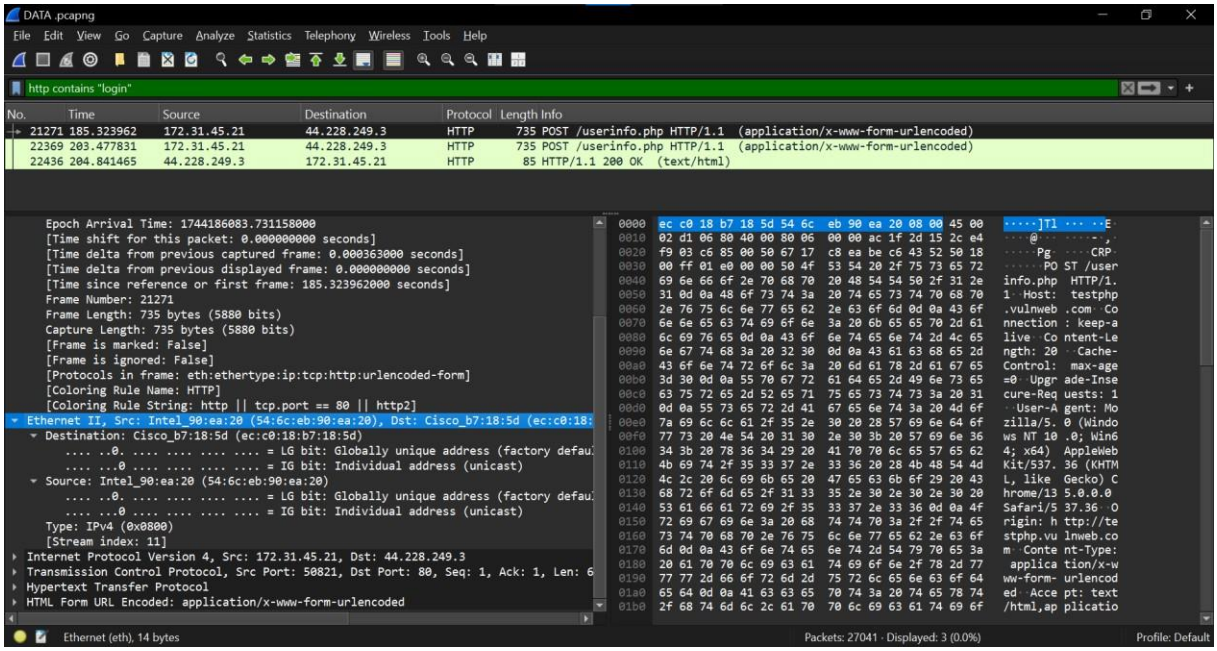
Phân tích theo từng tầng trong mô hình OSI.

Tầng 2: Data Link Layer (Tầng liên kết dữ liệu) – Ethernet Thông tin lấy được

Ethernet: Ethernet II

MAC nguồn (Source MAC) `54:6c:eb:90:ea:20`
MAC đích (Destination MAC) `0a:8e:dc:43:fb:64`
Loại giao thức `0x0800` – IPv4

Tầng này đảm bảo việc truyền dữ liệu giữa hai thiết bị trong cùng mạng LAN



Lớp 3: Network Layer (Tầng mạng) – IP

Thông tin lấy được:

IP nguồn (Source IP) `172.31.45.21`
IP đích (Destination IP) `44.228.249.3`
Giao thức lớp vận chuyển TCP
Gói tin được đóng gói theo chuẩn IPv4 và sử dụng TCP làm giao thức tầng vận chuyển.

Tầng 4: Transport Layer (Tầng giao vận) – TCP

Thông tin lấy được:

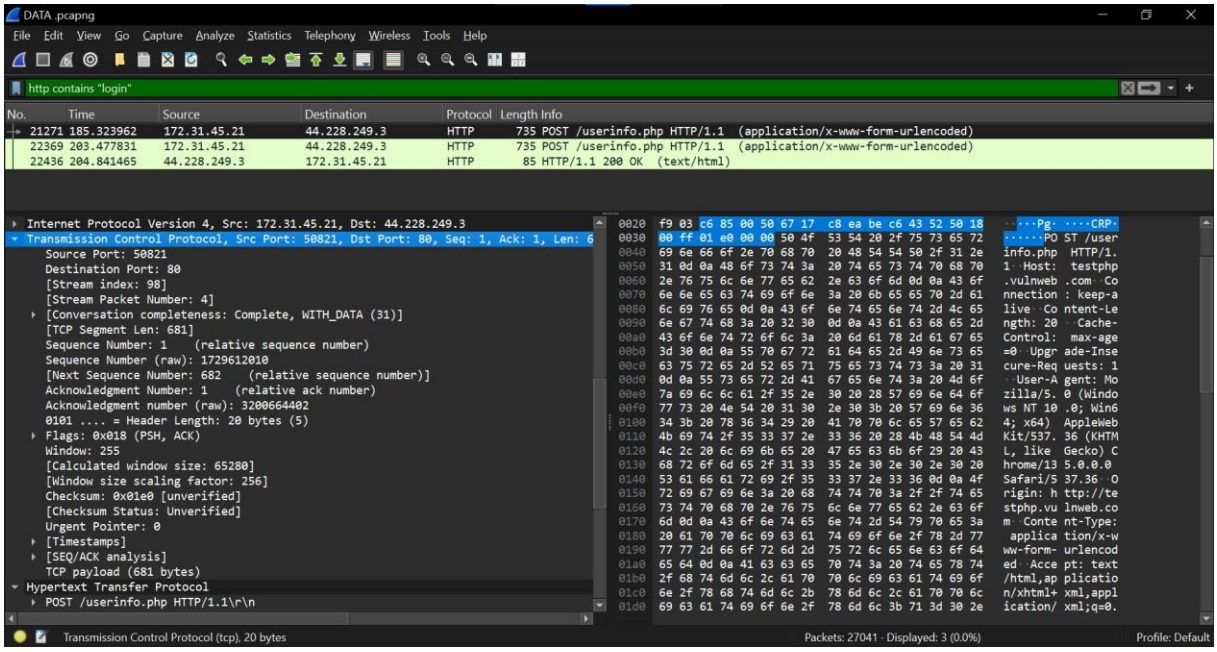
TCP: Cổng nguồn, cổng đích, số thứ tự (Sequence Number).

Port nguồn (Source Port)	`50821`
Port đích (Destination Port)	`80` (HTTP)
Số thứ tự (Sequence Number)	`1`

Tầng này đảm bảo gói tin được gửi đến đúng ứng dụng ở phía máy đích, có kiểm tra lỗi và gửi lại khi cần.

Tầng 7: Application Layer (Tầng ứng dụng) – HTTP

HTTP: Các phương thức (GET, POST), URL, Header,... Ví dụ đọc chi tiết một gói tin HTTP GET: Click đúp vào một gói tin HTTP GET → Mở rộng phần Hypertext Transfer Protocol. Kiểm tra các trường như: Host, User-Agent, Accept, Cookie.



Thông tin từ gói HTTP Request:

Phương thức HTTP: GET

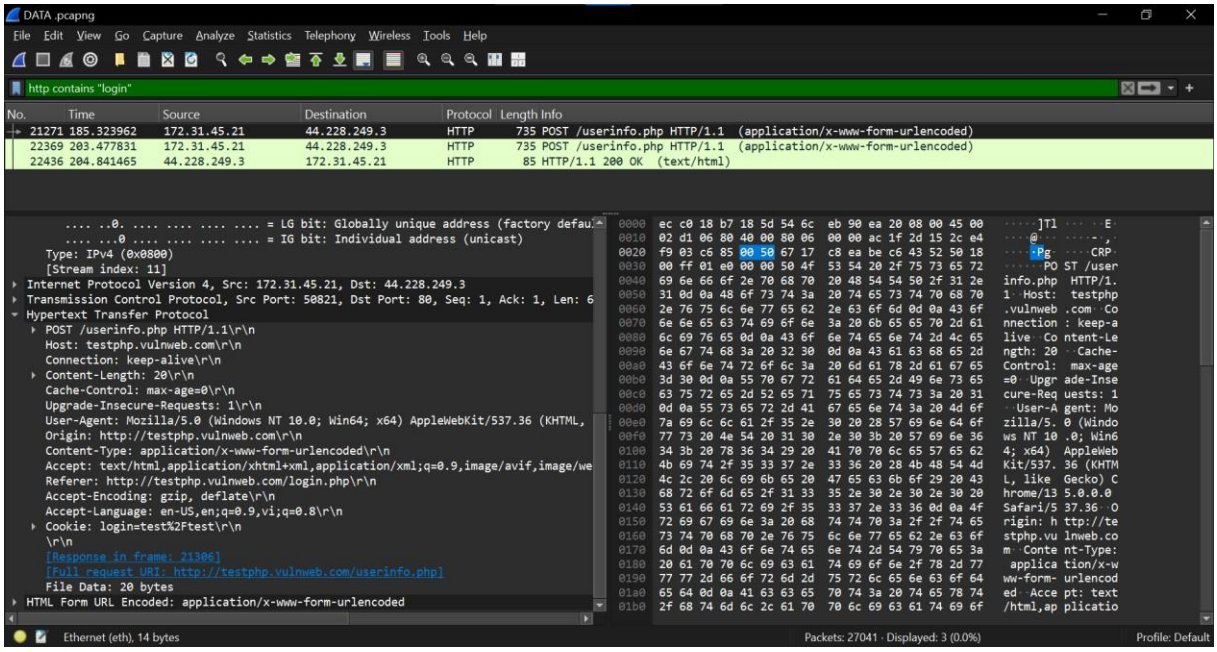
Host: www.google.com

Request URI: /search?q=test

Cookie: sessionid=abcd1234...

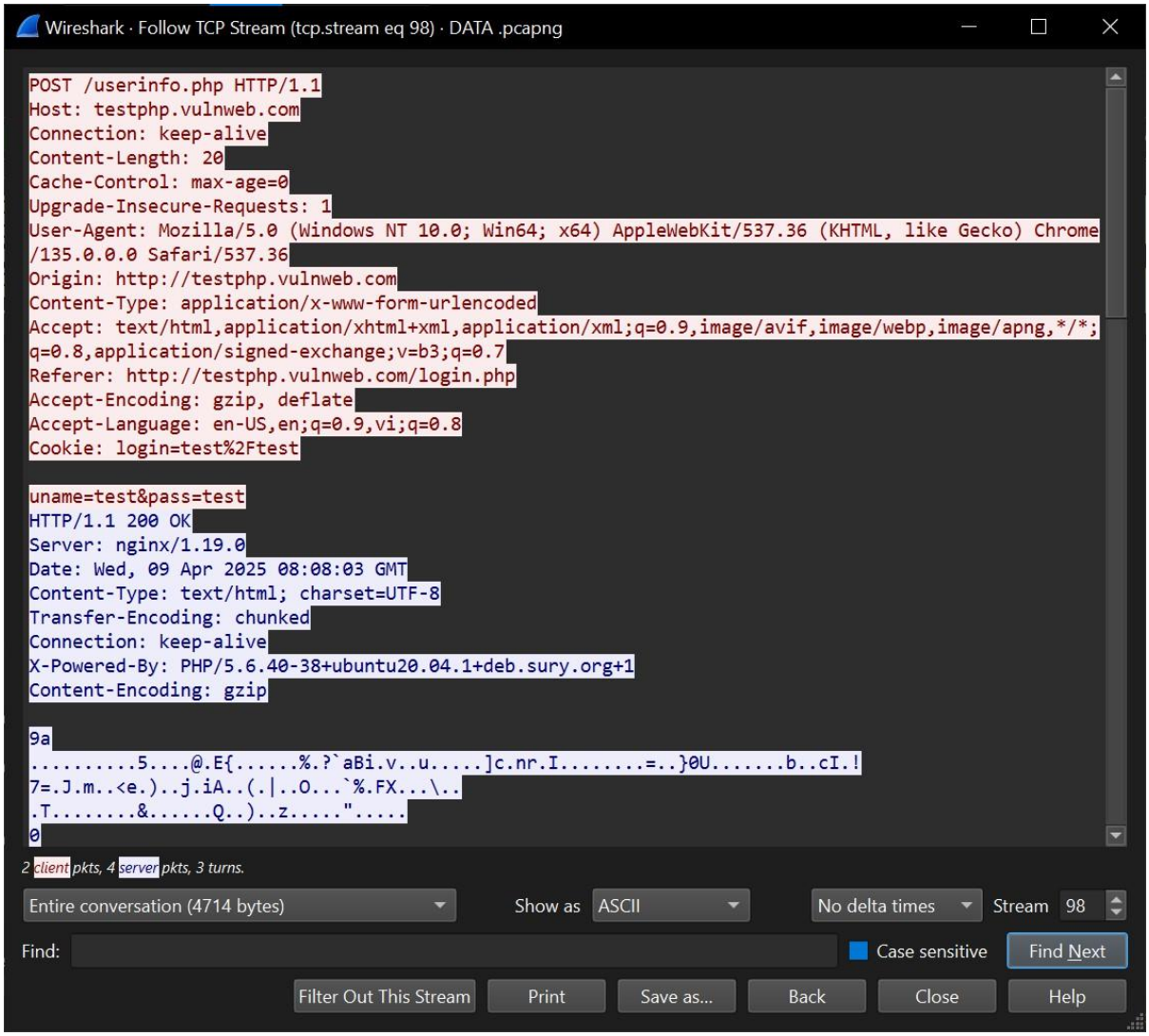
Payload: Không có trong GET (nếu là POST sẽ có dữ liệu)

Tầng này hiển thị dữ liệu người dùng thực sự sử dụng, ví dụ như truy cập web, gửi form, đăng nhập,...



Bước 5: Sử dụng tính năng Protocol Hierarchy hoặc Follow TCP Stream để quan sát toàn cục.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	63429	100.0	48363381	777 k	0	0	0	63429
Ethernet	100.0	63429	1.9	927202	14 k	0	0	0	63429
Internet Protocol Version 6	0.0	27	0.0	1080	17	0	0	0	27
User Datagram Protocol	0.0	18	0.0	144	2	0	0	0	18
Multicast Domain Name System	0.0	18	0.0	1476	23	18	1476	23	18
Internet Control Message Protocol v6	0.0	9	0.0	288	4	9	288	4	9
Internet Protocol Version 4	97.1	61571	2.5	1231420	19 k	0	0	0	61571
User Datagram Protocol	56.8	36003	0.6	288024	4629	0	0	0	36003
Simple Service Discovery Protocol	6.7	4223	3.2	1556310	25 k	4223	1556310	25 k	4223
QUIC IETF	38.2	24209	43.7	21127700	339 k	24209	21047032	338 k	24394
NetBIOS Name Service	0.0	6	0.0	300	4	6	300	4	6
Multicast Domain Name System	11.7	7426	2.8	1376978	22 k	7426	1376978	22 k	7426
Link-local Multicast Name Resolution	0.1	37	0.0	1049	16	37	1049	16	37
eXtensible Markup Language	0.0	21	0.0	13748	220	21	13748	220	21
Domain Name System	0.1	77	0.0	5510	88	77	5510	88	77
Data	0.0	4	0.0	3792	60	4	3792	60	4
Transmission Control Protocol	40.3	25562	1.2	560236	9003	19936	447716	7195	25562
Transport Layer Security	8.3	5267	33.7	16287643	261 k	5256	15058110	242 k	5390
Hypertext Transfer Protocol	0.0	11	0.0	5035	80	7	3675	59	11
Line-based text data	0.0	3	0.0	5851	94	3	5851	94	3
HTML Form URL Encoded	0.0	1	0.0	20	0	1	20	0	1
Data	0.5	331	0.3	162970	2619	331	162970	2619	331
Apache JServ Protocol v1.3	0.0	28	0.0	20781	333	28	20781	333	28
Internet Control Message Protocol	0.0	6	0.0	216	3	6	216	3	6
Address Resolution Protocol	2.9	1831	0.1	51268	823	1831	51268	823	1831



Bước 6:

```python
import pyshark

# Đường dẫn đến file pcapng
path = r'D:/MMT/mang_may_tinh/DATA.pcapng'
# Tạo đối tượng đọc file, lọc các gói HTTP Request
cap = pyshark.FileCapture(path, display_filter='http.request')

print("Phân tích gói HTTP REQUEST chứa từ khóa 'login' hoặc 'test'\n")

# Duyệt qua từng gói tin
for i, pkt in enumerate(cap):
    try:
        # Chuyển nội dung HTTP về chữ thường
        http_info = str(pkt).lower()

        if 'login' in http_info or 'test' in http_info:
            print("=" * 60)
            print(f"Gói #{i+1} có chứa từ khóa!")
```

```
            # Thời gian bắt gói
            print("Thời gian:", pkt.sniff_time)

            # ---------- TẦNG 2: Data Link Layer ----------
            if hasattr(pkt, 'eth'):
                print("MAC nguồn:", pkt.eth.src)
                print("MAC đích:", pkt.eth.dst)
                print("Loại giao thức tầng 2:", pkt.eth.type)
            else:
                print("Không có thông tin tầng 2 (ETH)")

            # ---------- TẦNG 3: Network Layer ----------
            if hasattr(pkt, 'ip'):
                print("IP nguồn:", pkt.ip.src)
                print("IP đích:", pkt.ip.dst)
                print("Giao thức tầng 3:", pkt.transport_layer)

                # Port nguồn và đích nếu là TCP hoặc UDP
                if pkt.transport_layer in ['TCP', 'UDP']:
                    layer = pkt[pkt.transport_layer.lower()]
                    print("Port nguồn:", layer.srcport)
                    print("Port đích:", layer.dstport)
            else:
                print("Không có thông tin tầng 3 (IP)")

            # ---------- HTTP Request (tầng 7) ----------
            if hasattr(pkt.http, 'request_method'):
                print("Phương thức HTTP:", pkt.http.request_method)

            if hasattr(pkt.http, 'host') and hasattr(pkt.http, 'request_uri'):
                print("URL:", f"http://{pkt.http.host}{pkt.http.request_uri}")

            if hasattr(pkt.http, 'cookie'):
                print("Cookie:", pkt.http.cookie)

            if hasattr(pkt.http, 'file_data'):
                print("Payload:", pkt.http.file_data)

        except Exception as e:
            print(f"Lỗi tại gói #{i+1}: {e}")
```