

# BÀI KIỂM TRA – PHIÊN TỔNG HỢP

## Báo Cáo Thực Hành

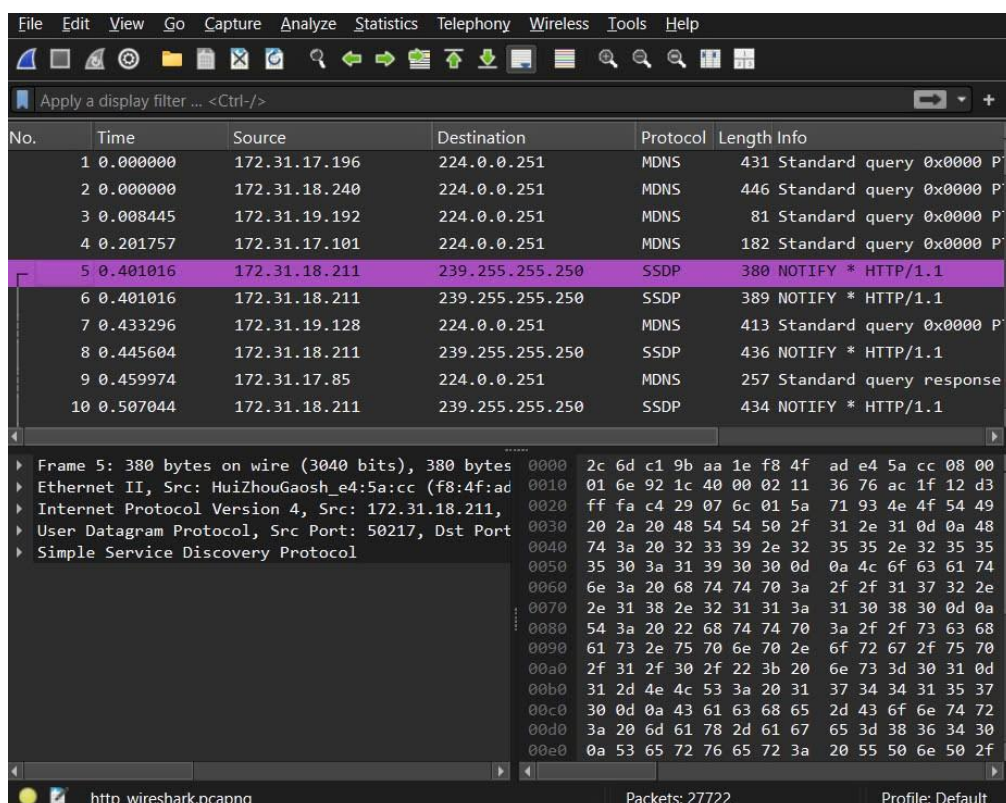
HỌ VÀ TÊN	MÃ SINH VIÊN
Nguyễn Quang Huy	22174600113
Nguyễn Thị Hà Mây	22174600031

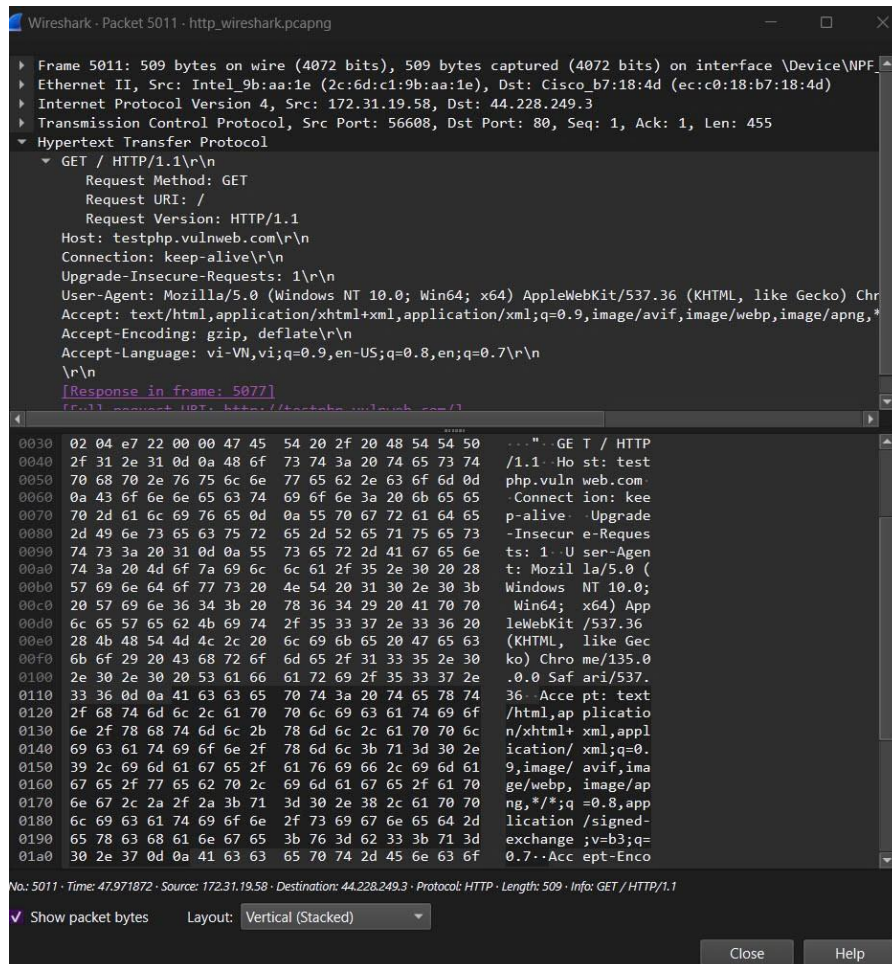
**Bước 1: Mở Wireshark, chọn card mạng, bắt gói khi truy cập một trang web.**

Khởi động Wireshark:

- Mở phần mềm Wireshark và chúng ta sẽ thấy danh sách các giao diện mạng (Network Interfaces) như: Ethernet, Wi-Fi, Loopback,...
- Chọn giao diện mà máy tính của người dùng đang sử dụng để kết nối mạng, ví dụ: Wi-Fi hoặc Ethernet.
- Bắt đầu thu thập gói tin:
  - Nhấn nút Start Capturing Packets (hình kính lúp màu xanh) để bắt đầu quá trình thu thập gói tin.
  - Chúng ta sẽ quan sát các gói tin chạy liên tục trên màn hình với các cột như: No. (Số thứ tự), Time, Source, Destination, Protocol, Length, Info.

**Bước 2: Lọc giao thức HTTP, truy cập một trang login, quan sát gói gửi dữ liệu.**





### Bước 3: Lưu file kết quả bắt gói (.pcapng).

```
import pyshark

import pandas as pd

import matplotlib.pyplot as plt

# Load the uploaded pcapng file
cap = pyshark.FileCapture("E:\Bài tập tin học cơ sở\http_wireshark.pcapng")

# Prepare a summary for layer 2 and layer 3
packet_data = []

for pkt in cap:
    try:
        packet_data.append({
            'No.': pkt.number,
```

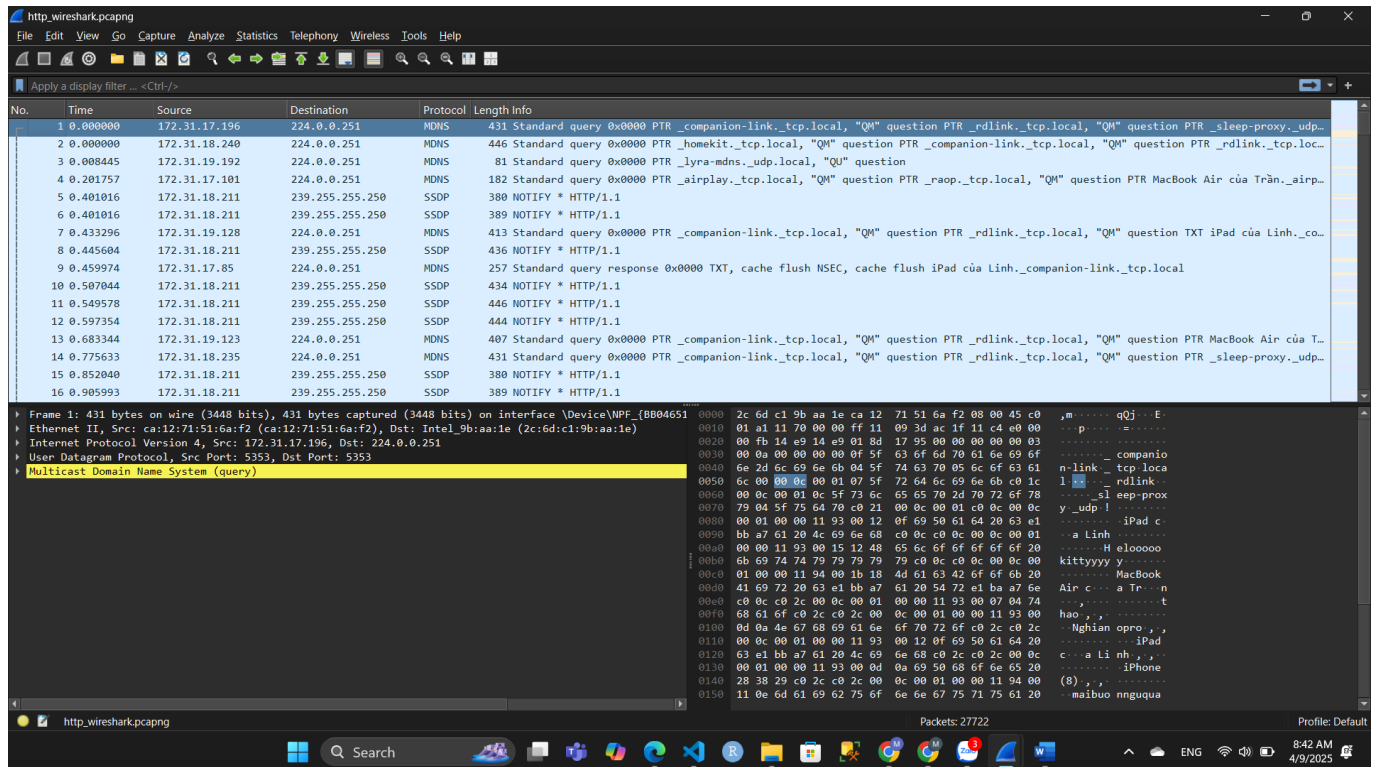
```
        'Time': pkt.sniff_time,
        'MAC Source': pkt.eth.src if hasattr(pkt, 'eth') else '',
        'MAC Destination': pkt.eth.dst if hasattr(pkt, 'eth') else '',
        'IP Source': pkt.ip.src if hasattr(pkt, 'ip') else '',
        'IP Destination': pkt.ip.dst if hasattr(pkt, 'ip') else '',
        'Protocol': pkt.highest_layer
    })
except Exception:
    continue

# Convert to DataFrame
df_packets = pd.DataFrame(packet_data)

# Plot: Protocol distribution (Protocol Hierarchy overview)
protocol_counts = df_packets['Protocol'].value_counts()

plt.figure(figsize=(10, 6))
protocol_counts.plot(kind='bar', color='skyblue')
plt.title('Protocol Hierarchy Overview')
plt.xlabel('Protocol')
plt.ylabel('Packet Count')
plt.xticks(rotation=45)
plt.tight_layout()
plt.grid(True)
plt.savefig('/mnt/data/protocol_hierarchy.png')

# Display first few packets as table
df_packets.head()
```



**Tên file:** http\_wireshark.pcapn

**Bước 4+ Bước 5 + Bước 6:**

**Bước 4: Mở lại file đã lưu, phân tích theo từng lớp trong mô hình OSI.**

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

**Bước 6: Viết mã Python dùng thư viện PyShark để truy xuất thông tin tầng 2 và tầng 3 từ file .pcapng.**

```
import pyshark

# Load the uploaded pcapng file
pcap_file_path = "E:\Bài tập tin học cơ sở\http_wireshark.pcapng"

# Use PyShark to read the file
cap = pyshark.FileCapture(pcap_file_path, use_json=True)

# Prepare a summary list for OSI layer analysis
packets_summary = []

for pkt in cap:
```

```
try:
    packet_info = {
        'No.': pkt.number,
        'Time': pkt.sniff_time.isoformat(),
        'MAC Source': pkt.eth.src if hasattr(pkt, 'eth') else 'N/A',
        'MAC Destination': pkt.eth.dst if hasattr(pkt, 'eth') else 'N/A',
        'IP Source': pkt.ip.src if hasattr(pkt, 'ip') else 'N/A',
        'IP Destination': pkt.ip.dst if hasattr(pkt, 'ip') else 'N/A',
        'Transport Layer': pkt.transport_layer if hasattr(pkt,
'transport_layer') else 'N/A',
        'Protocol': pkt.highest_layer
    }
    packets_summary.append(packet_info)
except Exception:
    continue

# Limit to first 10 packets for preview
packets_summary[:10]
```