

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN – ĐHQG.TPHCM
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN WIRESHARK
MÔN MẠNG MÁY TÍNH

| Giảng viên thực hành |
Thầy Nguyễn Thanh Quân

| Sinh viên thực hiện |
Võ Minh Khuê – 21120486
Hoàng Thị Khôn – 21120485
Nguyễn Ngọc Như Huyền – 21120475

Thành phố Hồ Chí Minh, tháng 11 năm 2022

I. ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH:

| Câu hỏi | Hoàn thành | Chưa hoàn thành |
|---------|-------------|-----------------|
| 1 | 100% | |
| 2 | 100% | |
| 3 | 100% | |
| 4 | 100% | |

II. TRẢ LỜI CÂU HỎI:

BÀI 1: PING

Câu 1:

- Địa chỉ IP host ping: 192.168.0.105
- Địa chỉ IP host được ping: 192.168.1.1

Câu 2: Không có thông tin port. Vì Ping sử dụng giao thức ICMP, giao thức này không cần thông tin port như giao thức TCP/UDP thay vào đó sử dụng Type và Code.

Câu 3:

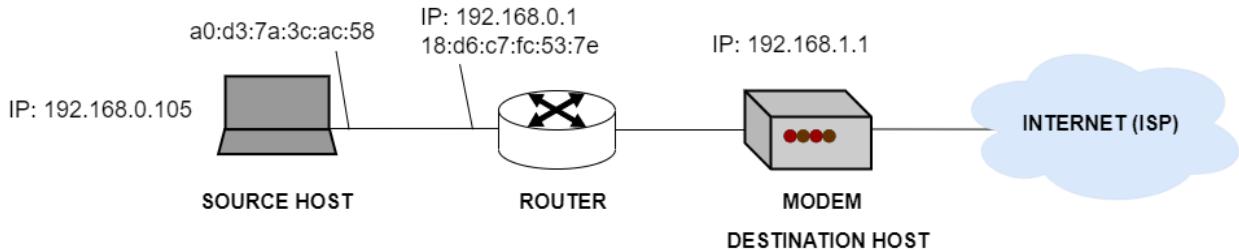
- a) Kích thước (bytes) của từng phần trong diagram:

| | | | |
|-----------|-------------|-----------|-----------------|
| 48 bytes | 8 bytes | 20 bytes | 14 bytes |
| ICMP data | ICMP header | IP header | Ethernet header |

- b) Có 2 gói tin ARP (gói tin truy vấn và gói tin phản hồi). Gói tin ARP được gửi đi để xác định địa chỉ MAC của cổng ra có IP 192.168.0.1. Ý nghĩa của các gói tin:

- Gói tin ARP thứ nhất (gói tin truy vấn) truy vấn tất cả các nút mạng khác trong đường mạng để tìm ra địa chỉ MAC của cổng ra có IP: 192.168.0.1.
- Sau khi tìm được nút mạng có địa chỉ IP: 192.168.0.1 thì nút mạng này tiến hành gửi gói tin ARP thứ hai (gói tin phản hồi) chứa thông tin địa chỉ MAC của nó cho nút nguồn để phía nguồn tiến hành đóng gói tin gửi cho phía đích.

- c) Vẽ sơ đồ logic của đường mạng:



BÀI 2: UDP

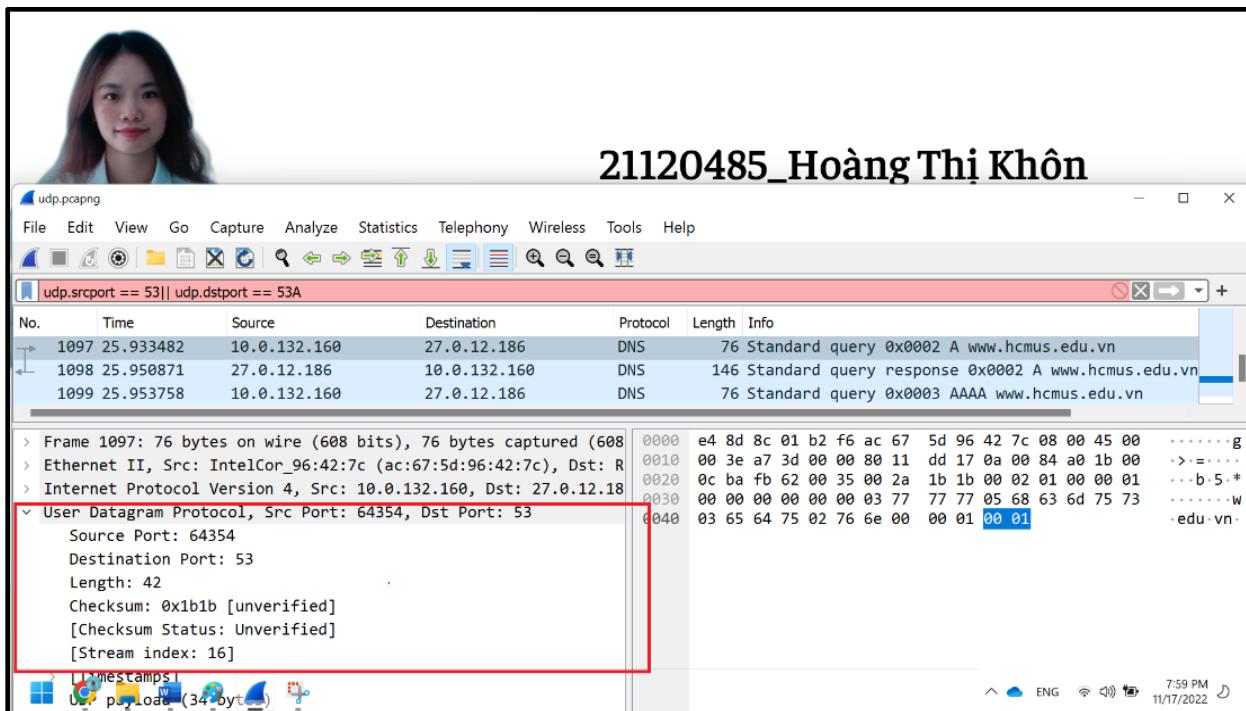
Câu 1:

Nslookup (viết tắt của name server lookup) là một chương trình tiện ích mạng được sử dụng để lấy thông tin về các máy chủ internet. Như tên của nó cho thấy, nó tìm thấy thông tin máy chủ tên cho các tên miền bằng cách truy vấn hệ thống tên miền (DNS).

“**Non-authoritative answer**” là không chứa các tệp nguồn gốc của vùng miền. Nghĩa là khi họ có một tệp bộ nhớ cache cho các miền được tạo từ tất cả các lần tra cứu DNS đã thực hiện trước đó. Nếu máy chủ DNS phản hồi cho một truy vấn DNS không có tệp gốc được gọi là “Non-authoritative answer”.

Câu 2: Có 4 trường thông tin trong phần header của gói tin UDP.

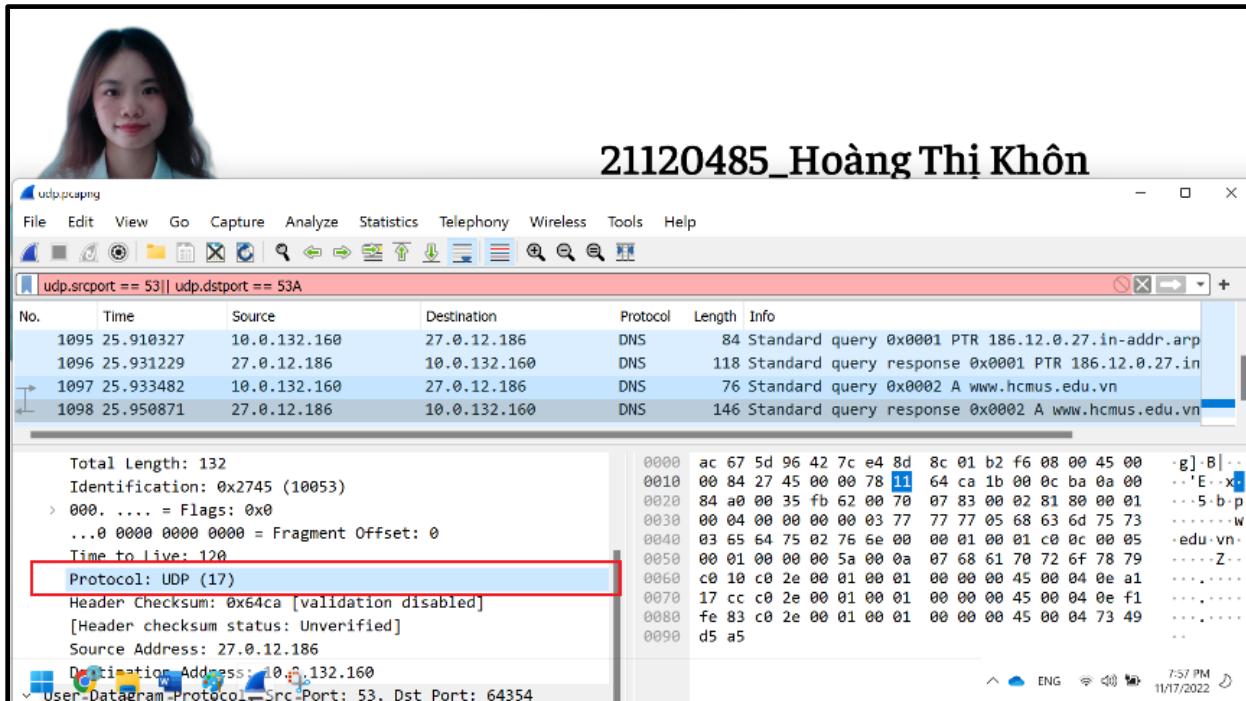
| Course port | Destination port | Datagram size(Length) | Checksum |
|-------------|------------------|-----------------------|----------|
| 2 bytes | 2 bytes | 2 bytes | 2 bytes |



Câu 3: Giá trị của trường Length: $8 + 72 = 80$ bytes. Trường Length là số bytes của UDP segment(bao gồm cả header và data).

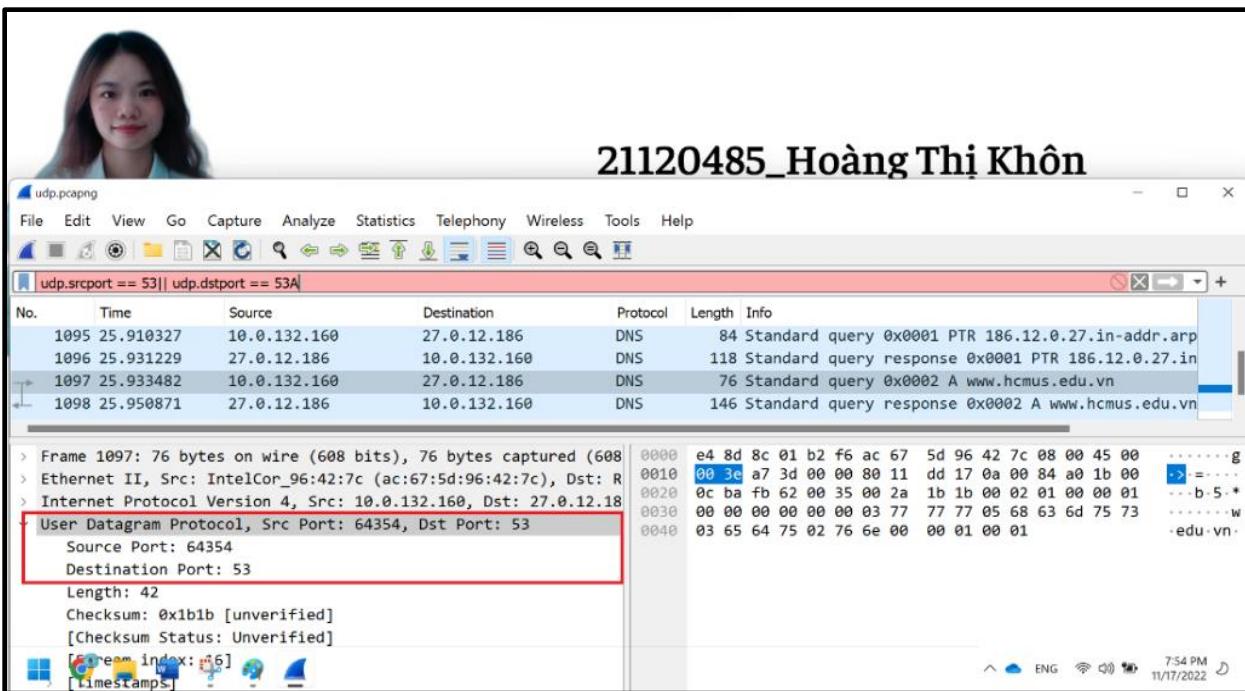
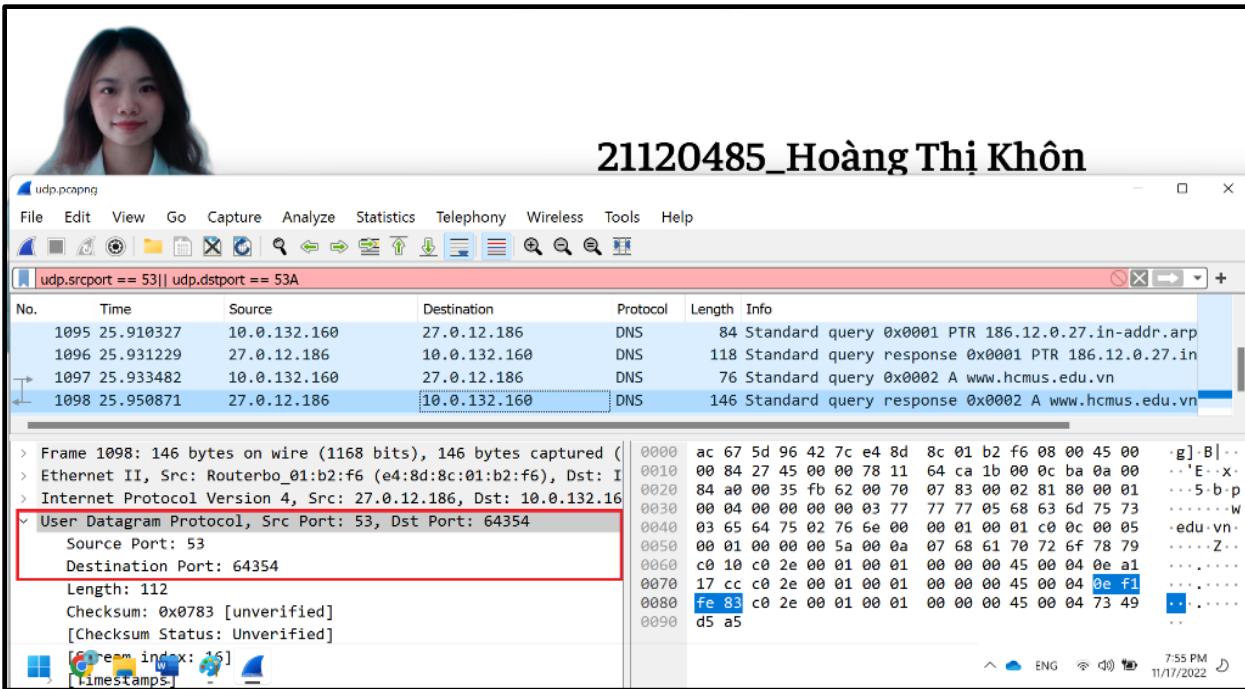
Câu 4: Protocol number của UDP: Protocol: UDP (17)

- Hộ 10: 17
- Hộ 16: 0x11



Câu 5: Số bytes lớn nhất mà payload (trừ đi 8 bytes của header) của UDP có thể chứa là $65535 - 8 = 65527$ bytes.

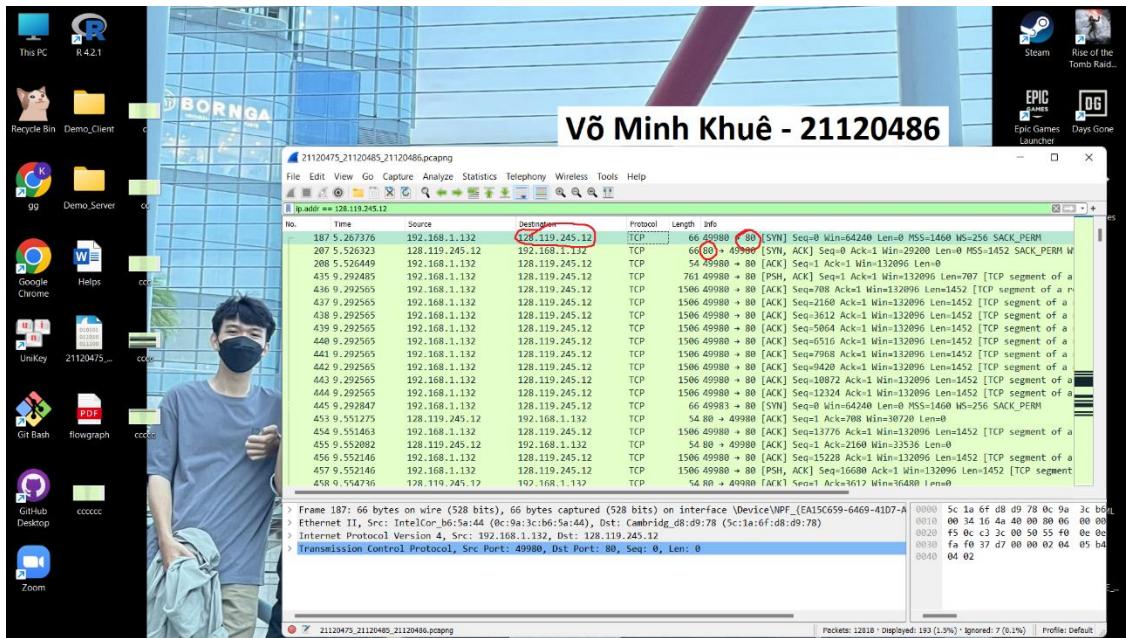
Câu 6: Source port của gói tin gửi đi cũng là destination port của gói tin phản hồi, ngược lại, source port của gói tin phản hồi là destination port của gói tin gửi đi.



BÀI 3: HTTP

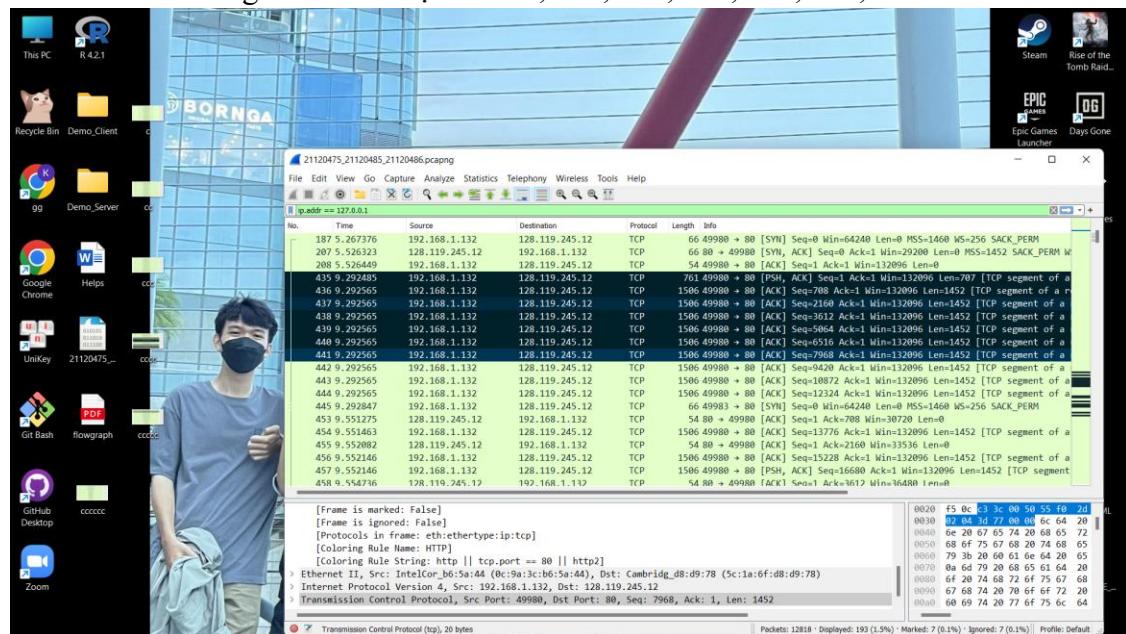
Câu 1:

- gaia.cs.umass.edu IP address: 128.119.245.12.
- Port dịch vụ được máy chủ dùng để gửi và nhận các gói tin TCP segment: port 80.



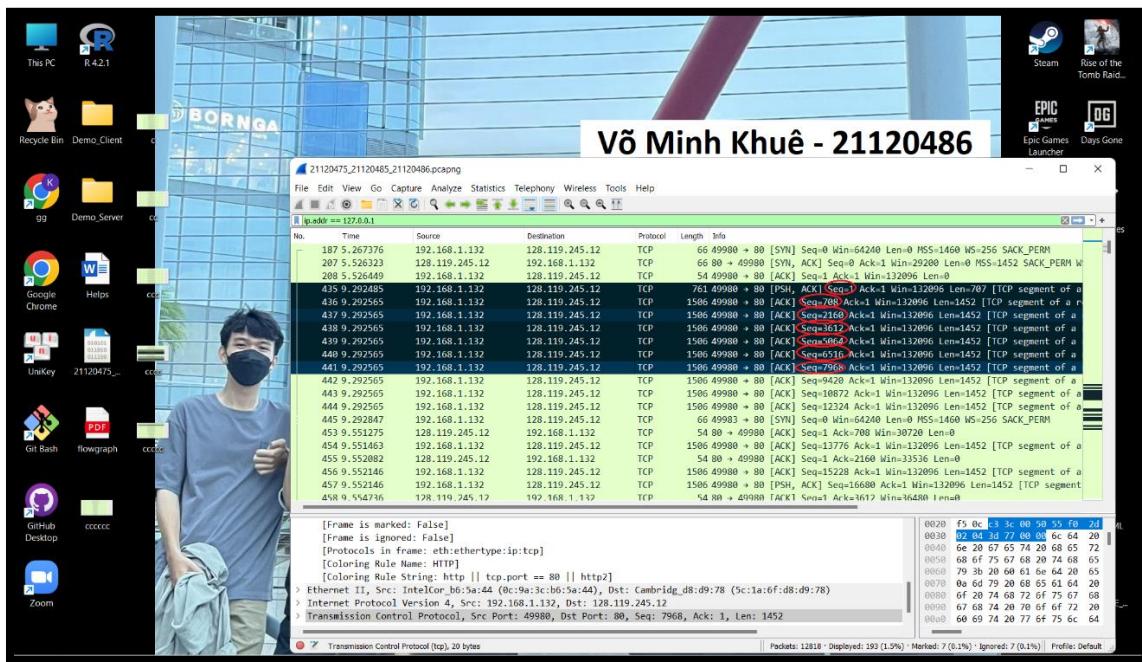
Câu 2:

- No. của 7 TCP segments lần lượt là: 435, 436, 437, 438, 439, 440, 441.



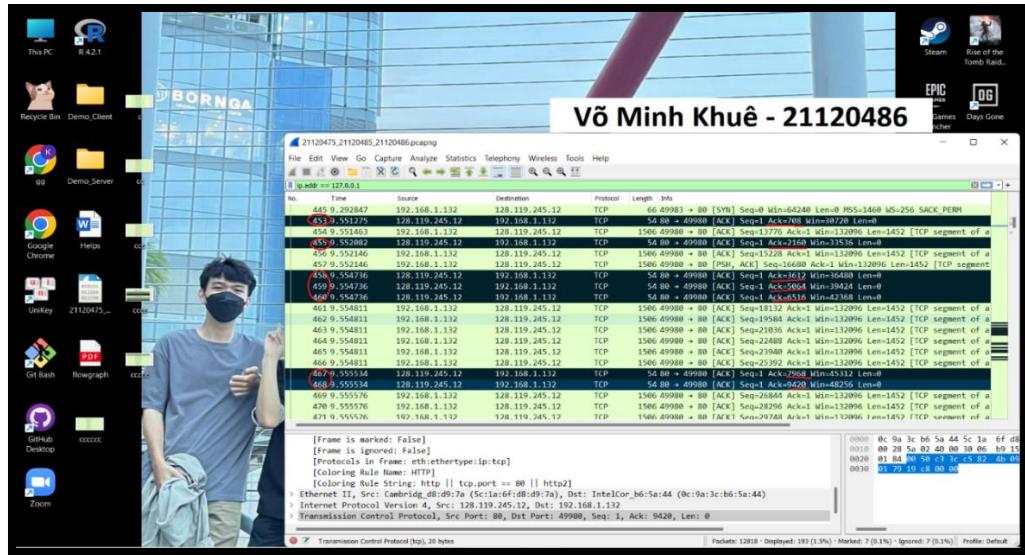
b. Sequence number của 7 TCP segments:

| No. | Sequence number |
|-----|-----------------|
| 435 | 1 |
| 436 | 708 |
| 437 | 2160 |
| 438 | 3612 |
| 439 | 5064 |
| 440 | 6516 |
| 441 | 7968 |



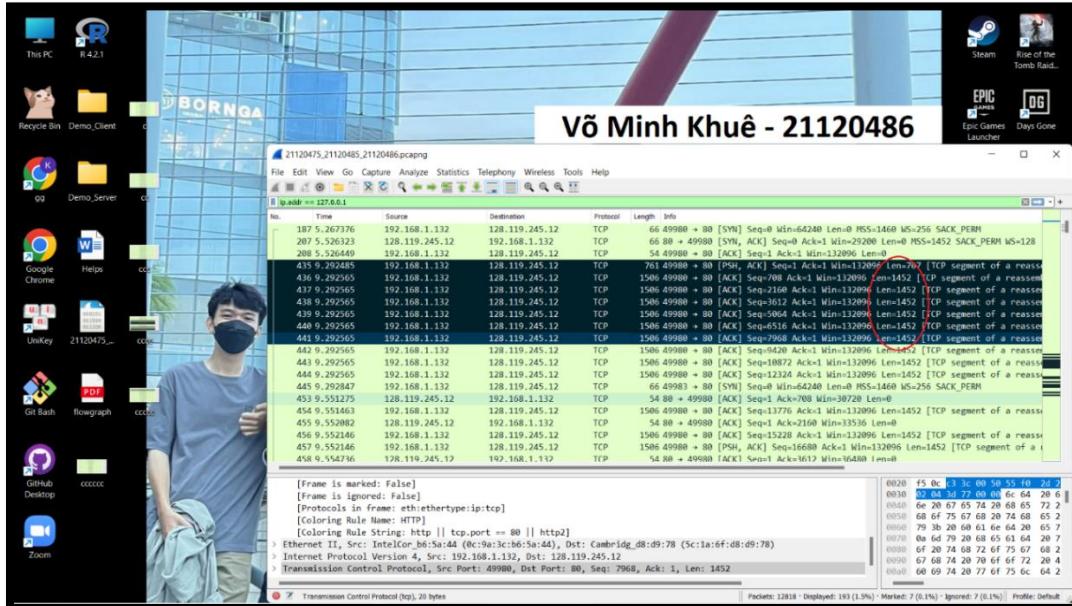
c.

| No. | No.Ack báo nhận |
|-----|-----------------|
| 435 | 453 |
| 436 | 455 |
| 437 | 458 |
| 438 | 459 |
| 439 | 460 |
| 440 | 467 |
| 441 | 468 |



d. Lượng data gửi trong mỗi TCP segment: (Hình minh họa ở câu a, dữ liệu “data” ở trường Len=...)

| No. | TCP segment data(bytes) |
|-----|-------------------------|
| 435 | 707 |
| 436 | 1452 |
| 437 | 1452 |
| 438 | 1452 |
| 439 | 1452 |
| 440 | 1452 |
| 441 | 1452 |



Câu 3:

Ta có máy chủ gaia gửi thông điệp 200 OK có ACK = 153029 nghĩa là đã có 153028 bytes đã được nhận (do có trừ Seq = 1). Hơn nữa, thời gian của thông điệp này là tại 10.614347s (packet #792) + thời gian bắt đầu gửi gói tin đầu tiên là tại 9.292485s (packet #435) ta có khoảng thời gian cho kết nối upload file này là 1.321862s nên ta được thông lượng (throughput) là:

$$\frac{153028 \text{ bytes}}{1.321862 \text{ seconds}} \approx 115767 \text{ (bytes per second).}$$

Câu 4:

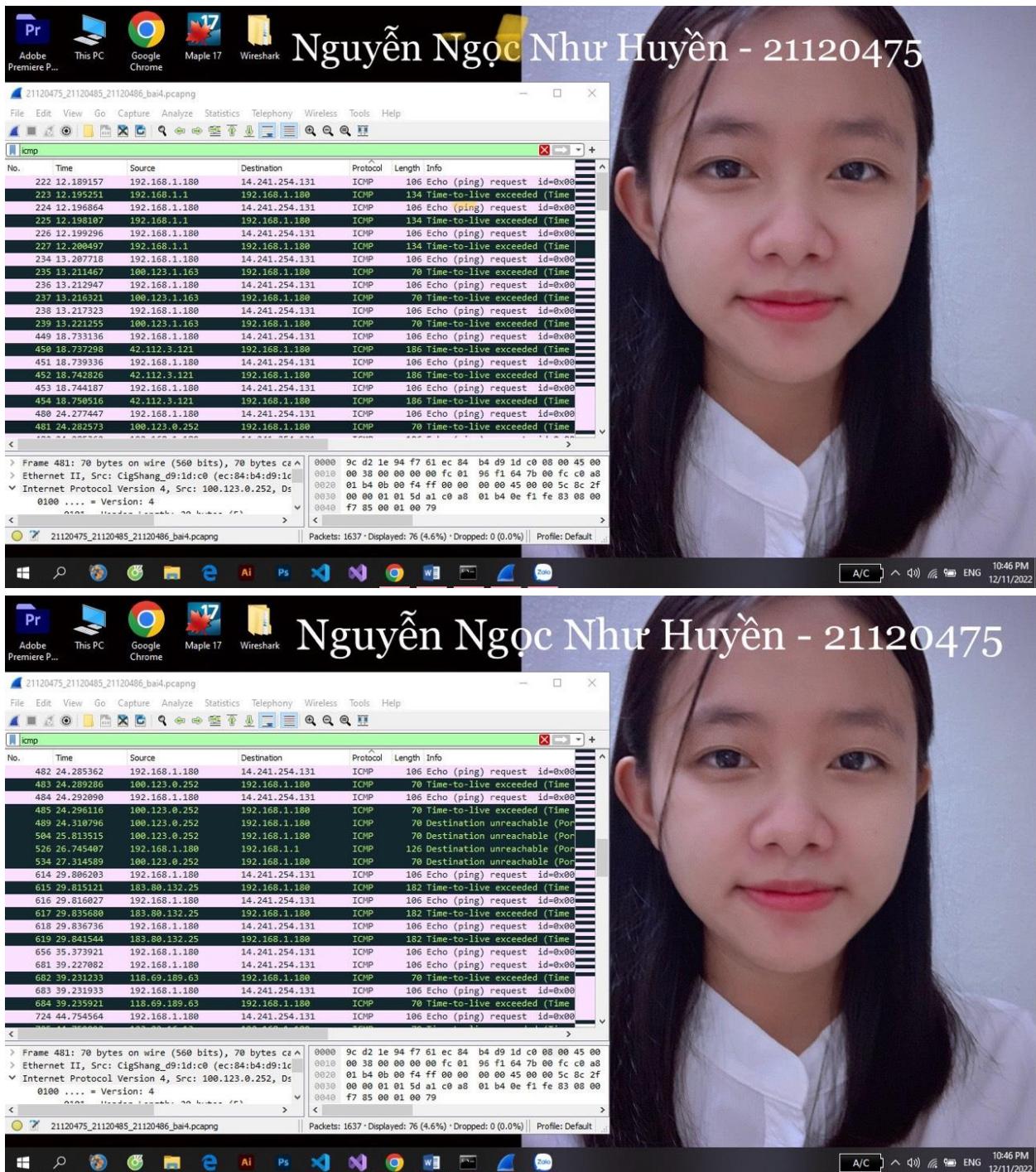
| Time | Source IP | Destination IP | Protocol | Sequence Number | Acknowledgment Number | Flags | Payload | Comment |
|-----------|-----------|----------------|---------------------|-----------------|-----------------------|------------|---|--|
| 9.819271 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=1 Ack=31200 Win=91904 Len=0 | TCP: 80 - 49980 [ACK] Seq=1 Ack=31200 Win=91 |
| 9.819271 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=1 Ack=34104 Win=97664 Len=0 | TCP: 80 - 49980 [ACK] Seq=1 Ack=34104 Win=91 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=467500 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=467500 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=468952 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=468952 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=70404 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=70404 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=73856 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=73856 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=73308 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=73308 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=76760 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=76760 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=79112 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=79112 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=77664 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=77664 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=79116 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=79116 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=80568 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=80568 Ack=1 Win=13 |
| 9.819323 | 49980 | 80 | TCP | 49980 | 80 | [PSH, ACK] | Seq=82030 Ack=1 Win=132096 Len=70 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [PSH, ACK] Seq=82030 Ack=1 Win=13 |
| 9.820655 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=1 Ack=37008 Win=103552 Len=0 | TCP: 80 - 49980 [ACK] Seq=1 Ack=37008 Win=10 |
| 9.820729 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=482727 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=482727 Ack=1 Win=13 |
| 9.820729 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=484179 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=484179 Ack=1 Win=13 |
| 9.820729 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=48621 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=48621 Ack=1 Win=13 |
| 9.820729 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=48783 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=48783 Ack=1 Win=13 |
| 9.821710 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=49134 Ack=1 Win=112236 Len=0 | TCP: 80 - 49980 [ACK] Seq=1 Ack=49134 Win=112236 Len=0 |
| 9.821768 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=49355 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=49355 Ack=1 Win=13 |
| 9.821768 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=49987 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=49987 Ack=1 Win=13 |
| 9.821768 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=91439 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=91439 Ack=1 Win=13 |
| 9.821768 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=92891 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=92891 Ack=1 Win=13 |
| 9.821768 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=94043 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=94043 Ack=1 Win=13 |
| 9.821768 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=95795 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=95795 Ack=1 Win=13 |
| 10.070702 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=1 Ack=44638 Win=115016 Len=0 | TCP: 80 - 49980 [ACK] Seq=1 Ack=44638 Win=115016 Len=0 |
| 10.070702 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=1 Ack=44638 Win=118016 Len=0 | TCP: 80 - 49980 [ACK] Seq=1 Ack=44638 Win=118016 Len=0 |
| 10.070774 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=97474 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=97474 Ack=1 Win=13 |
| 10.070774 | 49980 | 80 | TCP | 49980 | 80 | [PSH, ACK] | Seq=98639 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [PSH, ACK] Seq=98639 Ack=1 Win=13 |
| 10.070774 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=100151 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=100151 Ack=1 Win=13 |
| 10.070774 | 49980 | 80 | TCP | 49980 | 80 | [ACK] | Seq=101667 Ack=1 Win=132096 Len=1452 [TCP segment of a reassembled PDU] | TCP: 80 - 49980 [ACK] Seq=101667 Ack=1 Win=13 |
| 10.071702 | 49980 | 80 | [TCP Window Update] | 49980 | 80 | [ACK] | Seq=1 Ack=44638 Win=12960 SLE=45720 SRE=47172 | TCP: [TCP Window Update] 80 - 49980 [ACK] Seq=1 Ack=44638 Win=12960 SLE=45720 SRE=47172 |
| 10.074225 | 49980 | 80 | [TCP Window Update] | 49980 | 80 | [ACK] | Seq=1 Ack=44638 Win=123804 Len=0 SLE=45720 SRE=48624 | TCP: [TCP Window Update] 80 - 49980 [ACK] Seq=1 Ack=44638 Win=123804 Len=0 SLE=45720 SRE=48624 |
| 10.074225 | 49980 | 80 | [TCP Window Update] | 49980 | 80 | [ACK] | Seq=1 Ack=44638 Win=126948 Len=0 SLE=45720 SRE=50076 | TCP: [TCP Window Update] 80 - 49980 [ACK] Seq=1 Ack=44638 Win=126948 Len=0 SLE=45720 SRE=50076 |
| 10.074225 | 49980 | 80 | [TCP Window Update] | 49980 | 80 | [ACK] | Seq=1 Ack=44638 Win=129792 Len=0 SLE=45720 SRE=51528 | TCP: [TCP Window Update] 80 - 49980 [ACK] Seq=1 Ack=44638 Win=129792 Len=0 SLE=45720 SRE=51528 |

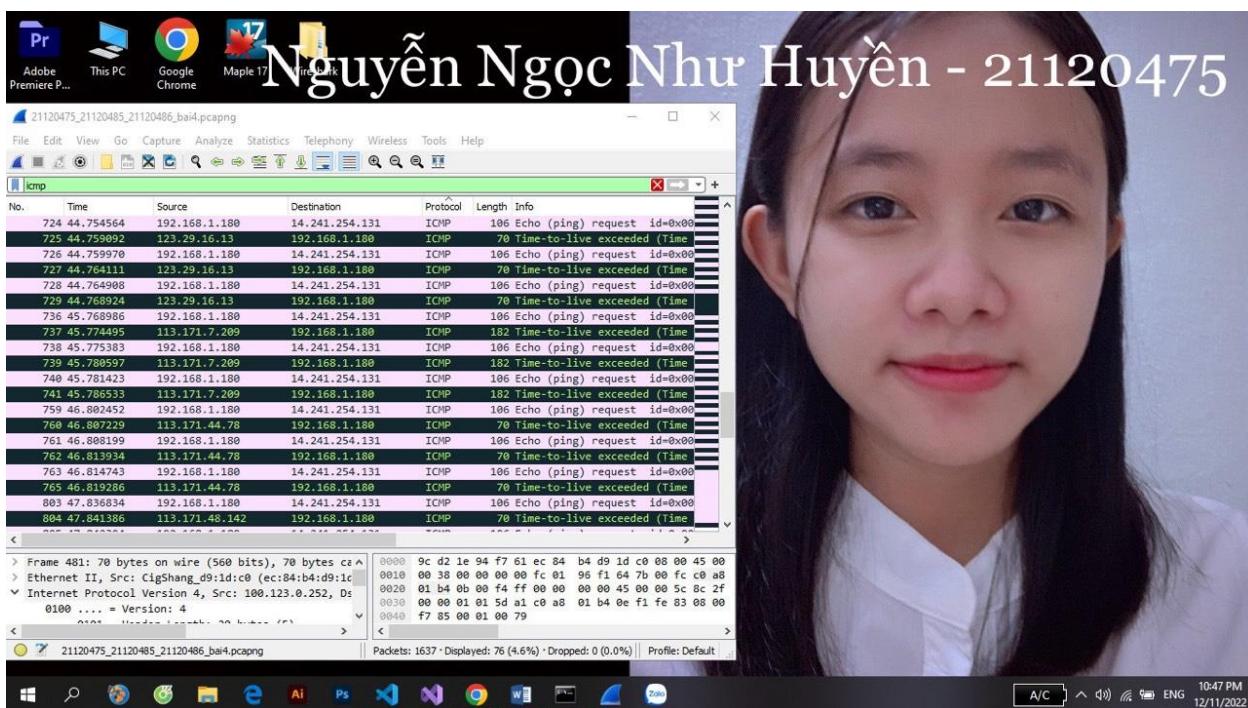
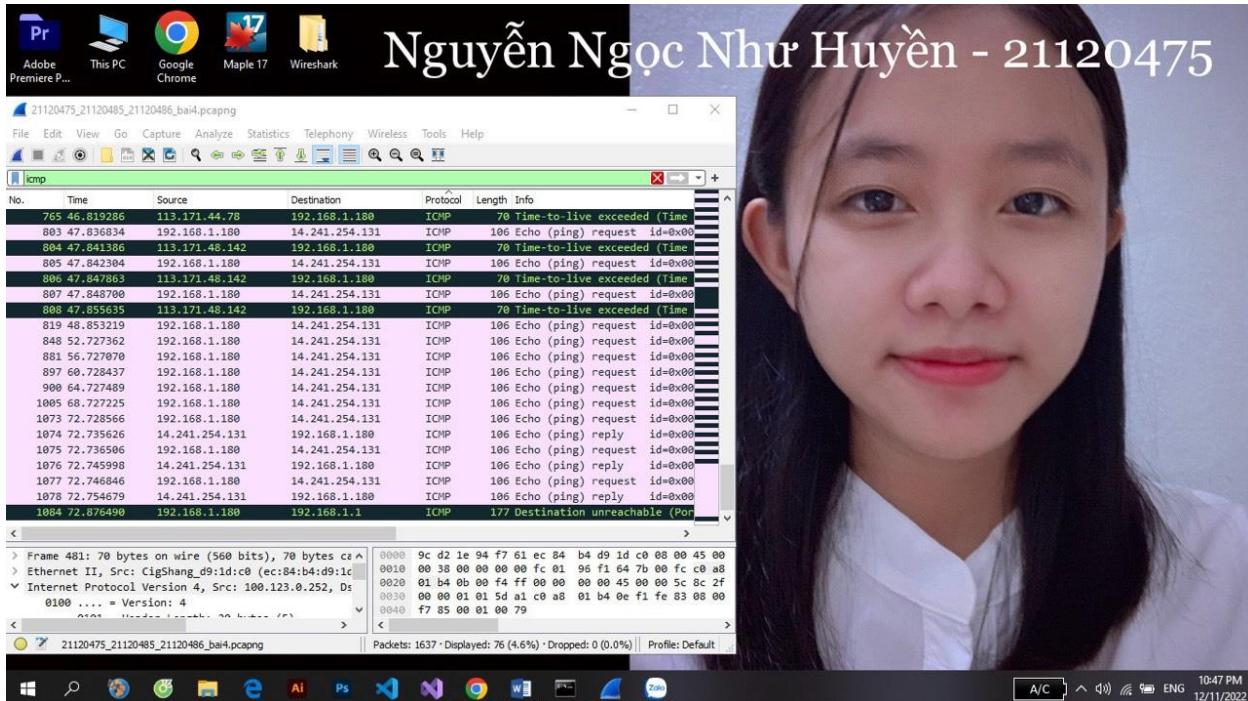
| Time | Source IP | Destination IP | Protocol | Content | Comment |
|-----------|-----------|----------------|---|--|---------|
| 10.332363 | 49980 | 80 | [ACK Seq=119027 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.332391 | 49980 | 80 | [ACK Seq=120479 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.332410 | 49980 | 80 | [ACK Seq=121931 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.339206 | 49980 | 80 | [TCP Dup ACK 55/15] 80 → 49980 [ACK Seq=1 Ack=44384 Win=166656 Len=1151 SLE=45720 SH=14505] | TCP: [TCP Dup ACK 55/15] 80 → 49980 [ACK Seq=1 Ack=44384 Win=166656 Len=1151 SLE=45720 SH=14505] | 80 |
| 10.339263 | 49980 | 80 | [ACK Seq=122383 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.340358 | 49980 | 80 | [TCP Dup ACK 55/15] 80 → 49980 [ACK Seq=1 Ack=44384 Win=166656 Len=1151 SLE=45720 SH=14505] | TCP: [TCP Dup ACK 55/15] 80 → 49980 [ACK Seq=1 Ack=44384 Win=166656 Len=1151 SLE=45720 SH=14505] | 80 |
| 10.340399 | 49980 | 80 | [ACK Seq=124915 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344078 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=105993 Win=145588 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=105993 Win=145588 Len=0] | 80 |
| 10.344120 | 49980 | 80 | [ACK Seq=126837 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344120 | 49980 | 80 | [ACK Seq=127774 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344120 | 49980 | 80 | [ACK Seq=129191 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344120 | 49980 | 80 | [PSH ACK Seq=130643 Ack=1 Win=13296 Len=1107] | TCP segment of a reassembled PDU | 80 |
| 10.344120 | 49980 | 80 | [ACK Seq=131760 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344120 | 49980 | 80 | [ACK Seq=131760 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344120 | 49980 | 80 | [ACK Seq=133024 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=134684 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=136136 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=137388 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=139693 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=140492 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=141944 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=143396 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.344230 | 49980 | 80 | [ACK Seq=144848 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.346544 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=107411 Win=173312 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=107411 Win=173312 Len=0] | 80 |
| 10.346544 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=103153 Win=173000 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=103153 Win=173000 Len=0] | 80 |
| 10.346599 | 49980 | 80 | [ACK Seq=146200 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.346599 | 49980 | 80 | [PSH ACK Seq=147175 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.346599 | 49980 | 80 | [ACK Seq=148104 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.347513 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=111767 Win=183216 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=111767 Win=183216 Len=0] | 80 |
| 10.347513 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=116123 Win=183321 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=116123 Win=183321 Len=0] | 80 |
| 10.347560 | 49980 | 80 | [ACK Seq=150656 Ack=1 Win=13296 Len=1452] | TCP segment of a reassembled PDU | 80 |
| 10.347560 | 49980 | 80 | POST /wreshark-labs/b3/b3-replay.htm HTTP/1.1 (text/plain) | HTTP POST /wreshark-labs/b3/b3-replay.htm HTTP/1.1 (text/plain) | 80 |
| 10.350278 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=117575 Win=183296 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=117575 Win=183296 Len=0] | 80 |
| 10.358777 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=120071 Win=183296 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=120071 Win=183296 Len=0] | 80 |
| 10.591212 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=120479 Win=183296 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=120479 Win=183296 Len=0] | 80 |
| 10.591212 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=121151 Win=183296 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=121151 Win=183296 Len=0] | 80 |
| 10.592357 | 49980 | 80 | 80 → 49980 [ACK Seq=1 Ack=122385 Win=183296 Len=0] | TCP: 80 → 49980 [ACK Seq=1 Ack=122385 Win=183296 Len=0] | 80 |

| Time | 192.168.1.132 | 128.119.245.12 | Comment |
|-----------|--|----------------|--|
| 10.344230 | 49980 → 80 [ACK] Seq=141944 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [ACK] Seq=141944 Ack=1 Win=1452 |
| 10.344230 | 49980 → 80 [ACK] Seq=143196 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [ACK] Seq=143196 Ack=1 Win=1452 |
| 10.344230 | 49980 → 80 [ACK] Seq=144486 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [ACK] Seq=144486 Ack=1 Win=1452 |
| 10.346544 | 80 → 49980 [ACK] Seq=1Ack=170411 Win=137312 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=170411 Win=137312 Len=0 |
| 10.346544 | 80 → 49980 [ACK] Seq=1Ack=170303 Win=137000 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=170303 Win=137000 Len=0 |
| 10.346599 | 49980 → 80 [ACK] Seq=148300 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [ACK] Seq=148300 Ack=1 Win=1452 |
| 10.346599 | 49980 → 80 [PSH, ACK] Seq=147751 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [PSH, ACK] Seq=147751 Ack=1 Win=1452 |
| 10.346599 | 49980 → 80 [ACK] Seq=149394 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [ACK] Seq=149394 Ack=1 Win=1452 |
| 10.347513 | 80 → 49980 [ACK] Seq=1Ack=11787 Win=130316 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=11787 Win=130316 Len=0 |
| 10.347513 | 80 → 49980 [ACK] Seq=1Ack=116321 Win=130332 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=116321 Win=130332 Len=0 |
| 10.347560 | 49980 → 80 [ACK] Seq=160656 Ack=1 Win=13296 Len=1452 [TCP segment of a reassembled POU] | 80 | TCP: 49980 → 80 [ACK] Seq=160656 Ack=1 Win=1452 |
| 10.347560 | POST /winshark/libw32-1/reply.htm HTTP/1.1 (text/plain) | 80 | HTTP: POST /winshark/libw32-1/reply.htm |
| 10.350278 | 80 → 49980 [ACK] Seq=1Ack=117578 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=117578 Win=13296 Len=0 |
| 10.350777 | 80 → 49980 [ACK] Seq=1Ack=118027 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=118027 Win=13296 Len=0 |
| 10.351212 | 80 → 49980 [ACK] Seq=1Ack=120479 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=120479 Win=13296 Len=0 |
| 10.351212 | 80 → 49980 [ACK] Seq=1Ack=121183 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=121183 Win=13296 Len=0 |
| 10.352357 | 80 → 49980 [ACK] Seq=1Ack=122380 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=122380 Win=13296 Len=0 |
| 10.357298 | 80 → 49980 [ACK] Seq=1Ack=124033 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=124033 Win=13296 Len=0 |
| 10.359592 | 80 → 49980 [ACK] Seq=1Ack=126327 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=126327 Win=13296 Len=0 |
| 10.602338 | 80 → 49980 [ACK] Seq=1Ack=127789 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=127789 Win=13296 Len=0 |
| 10.605520 | 80 → 49980 [ACK] Seq=1Ack=128183 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=128183 Win=13296 Len=0 |
| 10.605520 | 80 → 49980 [ACK] Seq=1Ack=131788 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=131788 Win=13296 Len=0 |
| 10.606221 | 80 → 49980 [ACK] Seq=1Ack=133232 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=133232 Win=13296 Len=0 |
| 10.607369 | 80 → 49980 [ACK] Seq=1Ack=134684 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=134684 Win=13296 Len=0 |
| 10.607369 | 80 → 49980 [ACK] Seq=1Ack=136138 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=136138 Win=13296 Len=0 |
| 10.607369 | 80 → 49980 [ACK] Seq=1Ack=137046 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=137046 Win=13296 Len=0 |
| 10.608571 | 80 → 49980 [ACK] Seq=1Ack=143396 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=143396 Win=13296 Len=0 |
| 10.613324 | 80 → 49980 [ACK] Seq=1Ack=146300 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=146300 Win=13296 Len=0 |
| 10.613324 | 80 → 49980 [ACK] Seq=1Ack=150863 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=150863 Win=13296 Len=0 |
| 10.614347 | 80 → 49980 [ACK] Seq=1Ack=153020 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=1Ack=153020 Win=13296 Len=0 |
| 10.614347 | HTTP/1.1 200 OK (text/html) | 80 | HTTP: HTTP/1.1 200 OK (text/html) |
| 10.657447 | 49980 → 80 [ACK] Seq=153029 Ack=778 Win=131238 Len=0 | 80 | TCP: 49980 → 80 [ACK] Seq=153029 Ack=778 Win=131238 Len=0 |
| 15.612172 | 80 → 49980 [FIN, ACK] Seq=779 Ack=153029 Win=13296 Len=0 | 80 | TCP: 80 → 49980 [FIN, ACK] Seq=779 Ack=153029 Win=13296 Len=0 |
| 15.612398 | 49980 → 80 [ACK] Seq=153029 Ack=779 Win=131238 Len=0 | 80 | TCP: 49980 → 80 [ACK] Seq=153029 Ack=779 Win=131238 Len=0 |
| 16.435311 | 49980 → 80 [FIN, ACK] Seq=153029 Ack=779 Win=131238 Len=0 | 80 | TCP: 49980 → 80 [FIN, ACK] Seq=153029 Ack=779 Win=131238 Len=0 |
| 16.695367 | 80 → 49980 [ACK] Seq=779 Ack=153030 Win=131296 Len=0 | 80 | TCP: 80 → 49980 [ACK] Seq=779 Ack=153030 Win=131296 Len=0 |

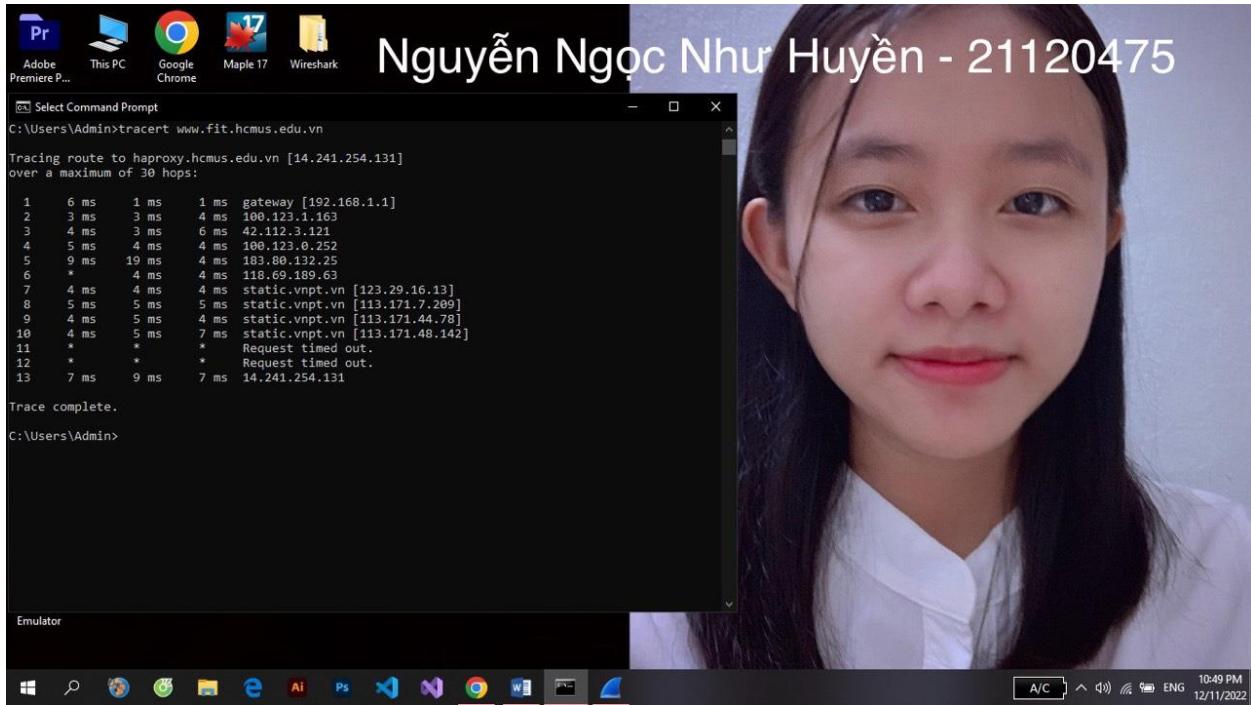
❖ BÀI 04: TRACEROUTE

Câu 1: Hình ảnh kết quả bát gói tin sau khi tracert





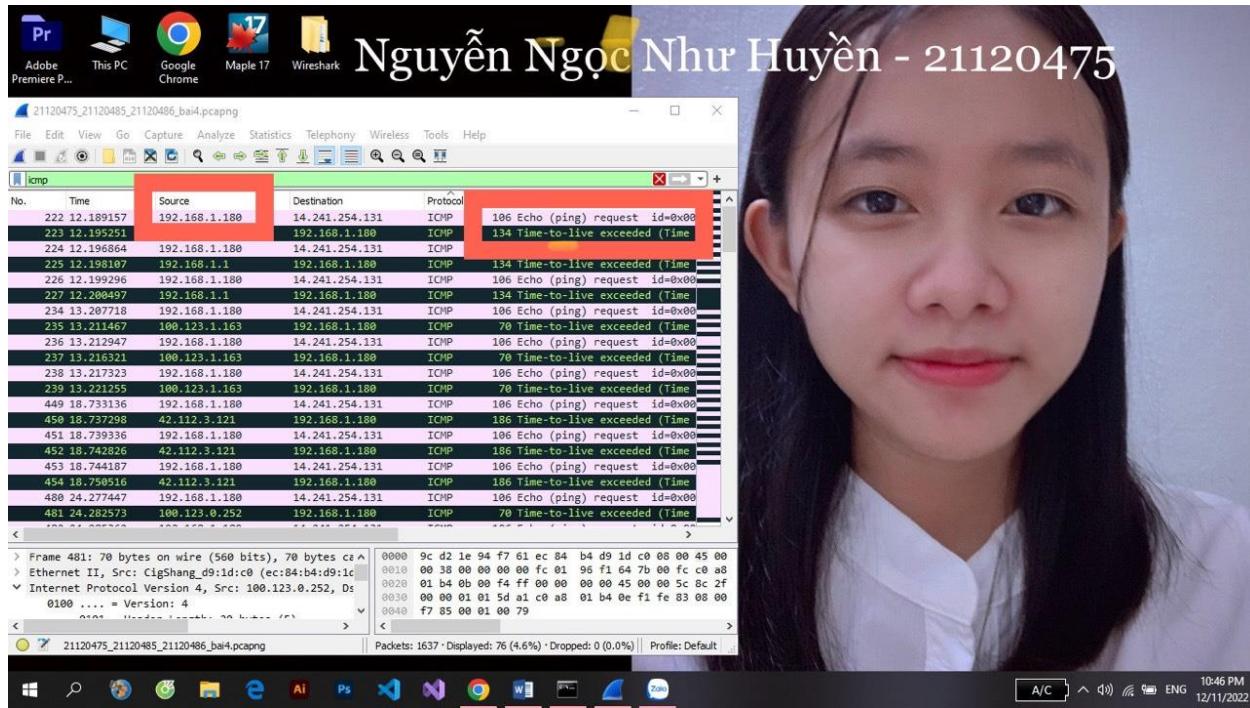
Command-line



Câu 2: Traceroute hoặc tracert là một công cụ dòng lệnh dùng để xác định đường đi từ nguồn tới đích của một gói giao thức mạng Internet (IP–Internet protocol) bao gồm:

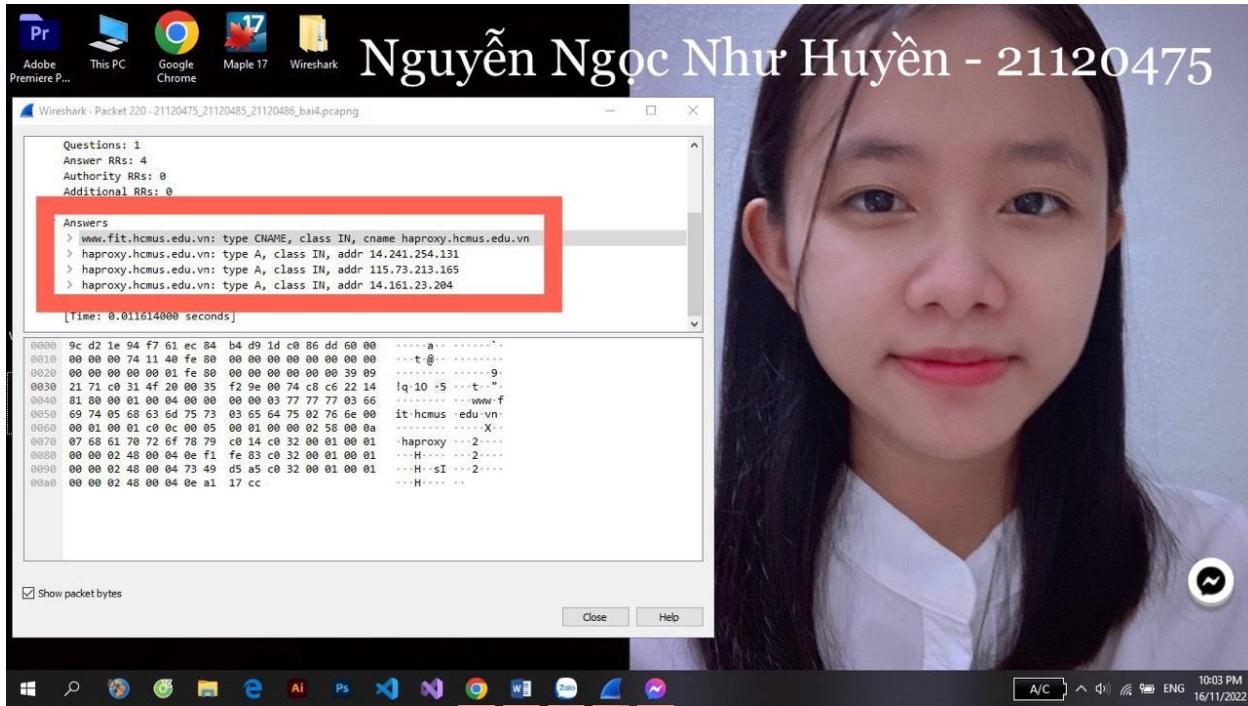
- Xác định IP của các router (hop) mà gói tin đi qua.
- Xác định thời gian trung bình từ lúc host nguồn gửi gói tin đi đến lúc nhận phản hồi tại các router hoặc host đích (RTT–Round trip time).
- Xác định nơi mà gói tin không thể gửi đến (nơi xảy ra lỗi: host, net hoặc port).

Câu 3: Địa chỉ IP của máy gửi request là: 192.168.1.180



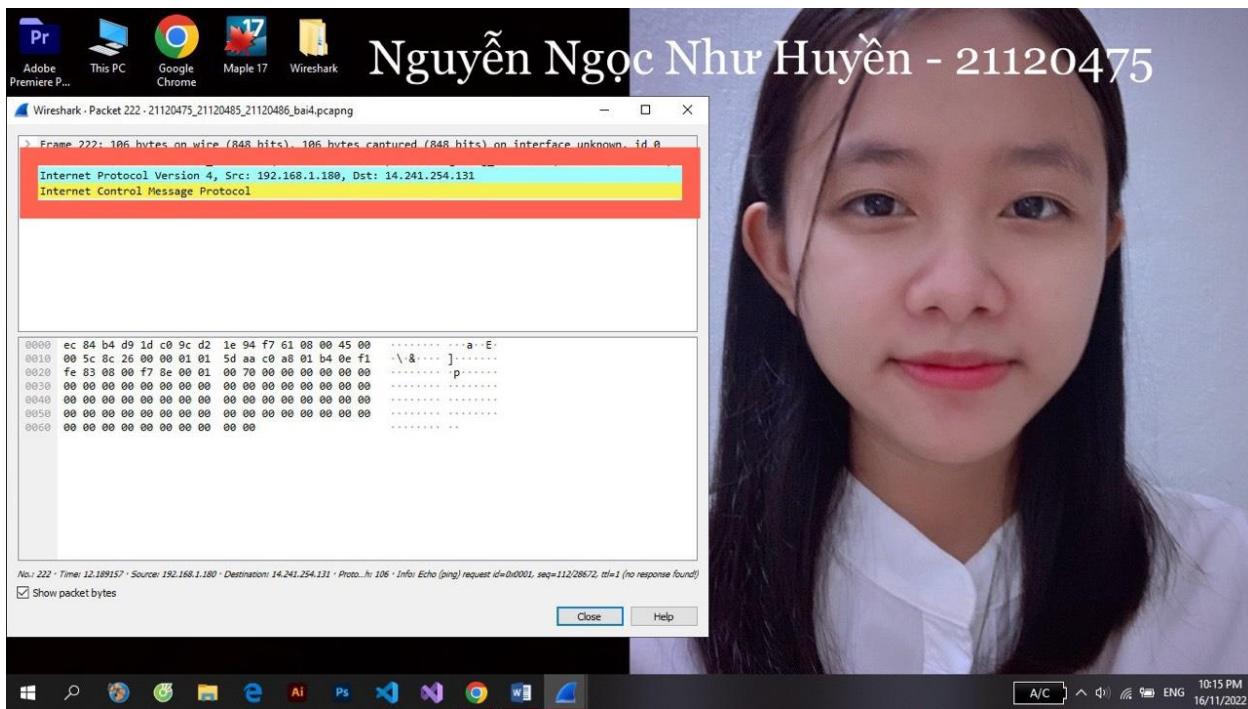
Máy tính gửi gói tin request đến host đích nên IP nguồn là IP của máy gửi request.

Câu 4: Máy tính xác định được địa chỉ IP của FIT nhờ giao thức ở tầng ứng dụng là DNS (Domain Name System). Đầu tiên, máy tính gửi gói tin truy vấn tên miền đến máy chủ DNS cục bộ của mình. Máy chủ DNS cục bộ này tiếp tục gửi gói tin truy vấn đến máy chủ DNS của FIT và nhận gói tin response trả về địa chỉ IP cho máy tính như hình sau:



Câu 5:

- a. Protocol được sử dụng của những gói tin sau đó là ICMP và IPv4.

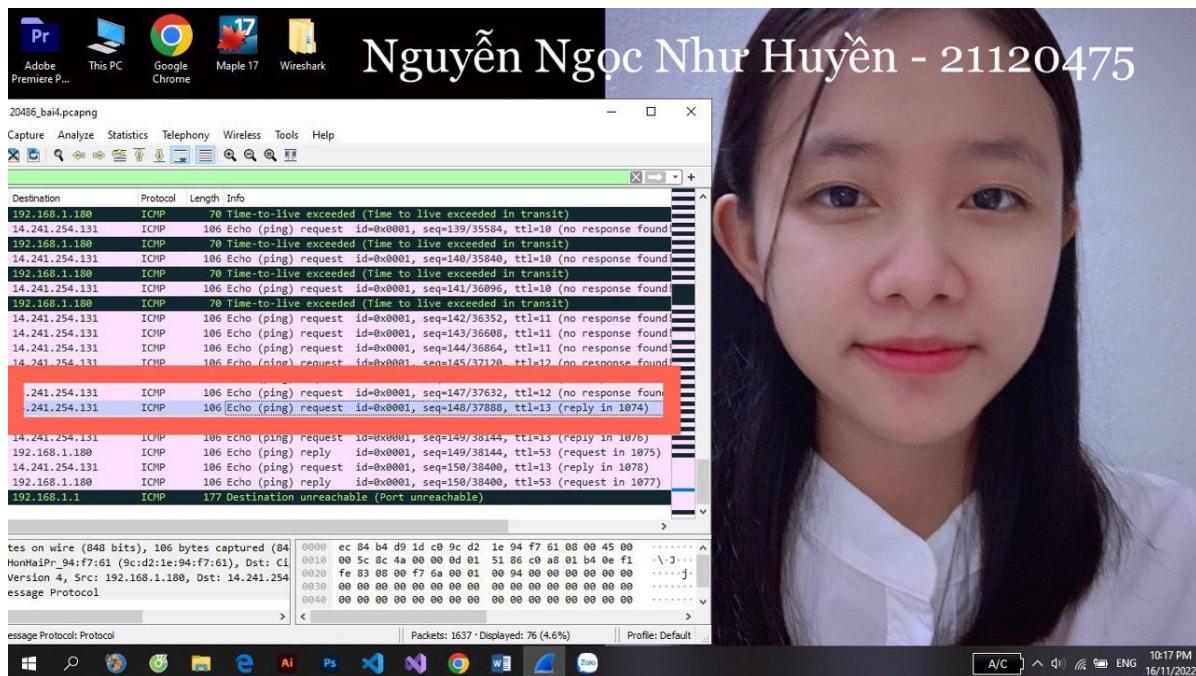


- b. Có 37 gói tin được gửi đi (request) trước khi nhận được phản hồi đầu tiên cho những request.

Sau khi máy tính gửi gói tin có TTL = 13 thì nhận được phản hồi đầu tiên. Trước đó, ứng với mỗi TTL máy tính gửi 3 gói tin nên số gói tin được gửi đi trước khi nhận được phản hồi đầu tiên là: $12 \times 3 + 1 = 37$.

c. TTL của gói tin cuối cùng được gửi trước khi nhận được gói tin phản hồi đầu tiên cho những gói tin request là 13.

Không thể thấy thông tin port trong các gói tin gửi đi. Vì giao thức được gửi kèm trong các gói tin IP (IP datagram) là ICMP. Giao thức này không cần thông tin port như giao thức TCP/UDP thay vào đó sử dụng Type và Code.



d. Gói tin phản hồi đầu tiên là trả lời cho gói tin request thứ 37.

III. BẢNG PHÂN CÔNG CÔNG VIỆC:

| Công việc | Người thực hiện |
|-----------|-----------------------|
| Câu 1 | Cả nhóm |
| Câu 2 | Hoàng Thị Khôn |
| Câu 3 | Võ Minh Khuê |
| Câu 4 | Nguyễn Ngọc Như Huyền |

IV. TÀI LIỆU THAM KHẢO:

1. Giáo trình:

+ **Mạng máy tính**, Mai Văn Cường - Trần Trung Dũng - Trần Hồng Ngọc - Lê Ngọc Sơn - Lê Giang Thanh - Trương Thị Mỹ Trang - Đào Anh Tuấn, NXB Khoa học & Kỹ Thuật.

+ *Computer Networking: A top-down approach featuring the Internet*, 7th edition, James F. Kurose, Keith W. Ross.

2. Website: