

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

----------



NHẬP MÔN MẠNG MÁY TÍNH – IT005.Q111.1

BÁO CÁO THỰC HÀNH LAB 3

Phân tích hoạt động giao thức TCP - UDP

Giảng viên hướng dẫn: Nguyễn Thanh Nam

Sinh viên thực hiện: Đặng Trường Sinh – 24521524

MỤC LỤC

| | |
|---|-----------|
| 1. Task 1: Phân tích hoạt động giao thức UDP | 4 |
| 1.1 Streaming Video sử dụng UDP | 4 |
| 1.2 Tiến hành bắt gói tin UDP khi streaming video | 6 |
| 1.3 Phân tích hoạt động giao thức UDP..... | 8 |
| 2. Task 2: Phân tích hoạt động giao thức TCP..... | 14 |
| 2.1 Streaming video sử dụng HTTP và bắt gói tin TCP..... | 14 |
| 2.2 Phân tích hoạt động giao thức TCP | 17 |

DANH MỤC HÌNH ẢNH

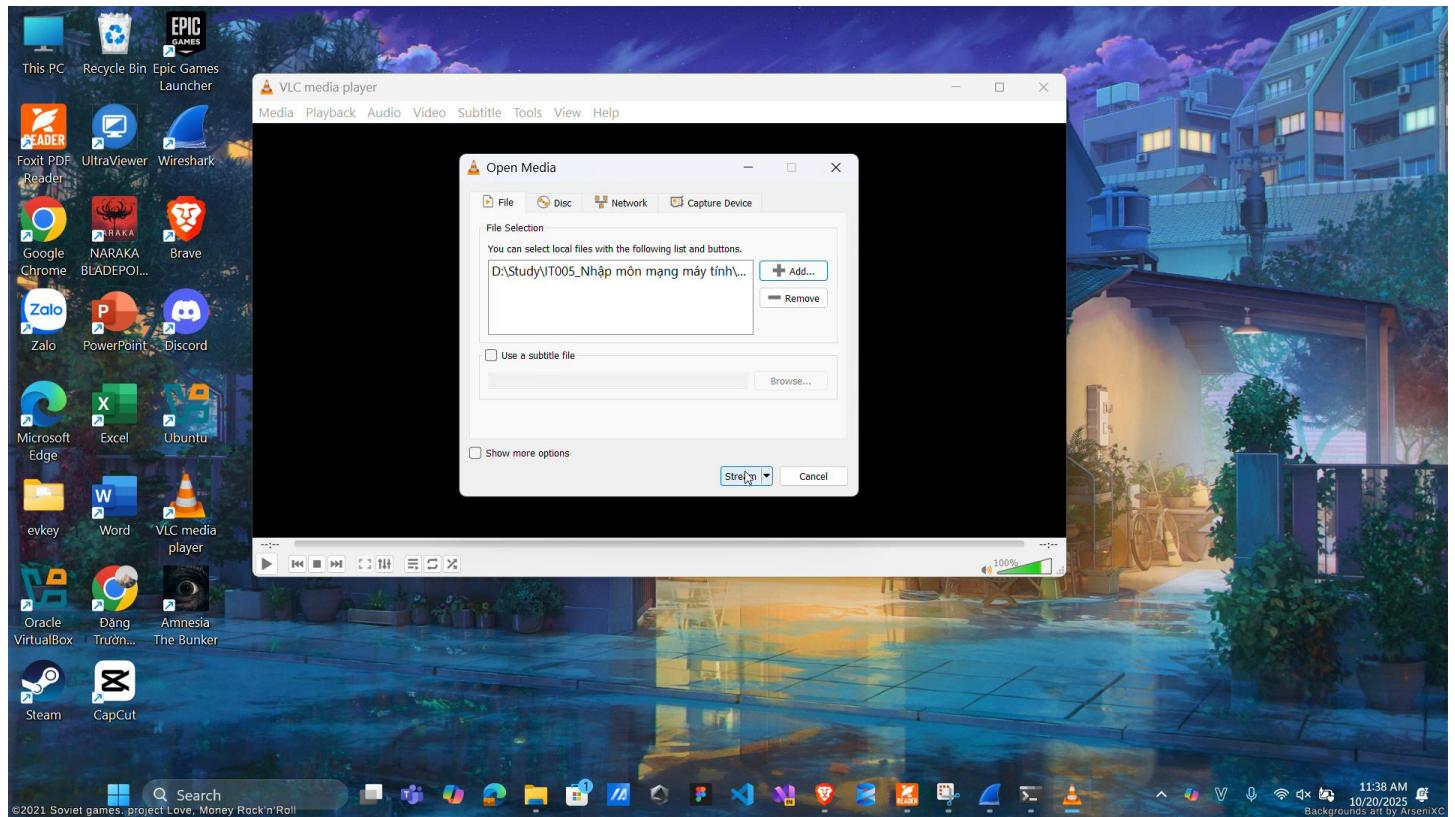
| | |
|---|----|
| Hình 1. Chọn video cần stream | 4 |
| Hình 2. Chọn RTSP và Port mặc định khi stream là 8554 | 5 |
| Hình 3. Bắt đầu stream | 5 |
| Hình 4. Truy cập để xem video stream từ máy của sinh viên khác | 6 |
| Hình 5. Xem video 1 thời gian ngắn..... | 7 |
| Hình 6. Các trường có trong UDP header..... | 8 |
| Hình 7. Độ dài của trường Source Port | 9 |
| Hình 8. Độ dài của trường Destination Port | 9 |
| Hình 9. Độ dài của trường Length..... | 10 |
| Hình 10. Độ dài của trường Checksum | 10 |
| Hình 11. Độ dài của UDP payload..... | 11 |
| Hình 12. Độ dài của IP header..... | 12 |
| Hình 13. Gói tin máy mình gửi..... | 13 |
| Hình 14. Gói tin phản hồi | 13 |
| Hình 15. Chọn video để stream | 14 |
| Hình 16. Chọn HTTP với Port mặc định của là 8080..... | 15 |
| Hình 17. Truy cập để xem video stream của sinh viên khác | 16 |
| Hình 18. Xem video 1 thời gian ngắn..... | 16 |
| Hình 19. Địa chỉ IP và TCP Port của Client | 17 |
| Hình 20. Địa chỉ IP và TCP Port của Sever | 18 |
| Hình 21. Gói tin TCP có cờ SYN | 19 |
| Hình 22. Sequence number và giá trị Acknowledgement trong SYN/ACK segment | 20 |
| Hình 23. Nhận biết SYN/ACK segment..... | 21 |
| Hình 24. 6 segment đầu tiên mà Sever gửi cho Client | 22 |
| Hình 25. Sequence Number của Segment 30. | 23 |
| Hình 26. Sequence Number của Segment 33 | 23 |
| Hình 27. Sequence Number của Segment 35 | 24 |
| Hình 28. Sequence Number của Segment 37 | 24 |
| Hình 29. Sequence Number của Segment 38 | 25 |
| Hình 30. Sequence Number của Segment 40 | 25 |
| Hình 31. Biểu đồ Sequence Number (Stevens) | 26 |
| Hình 32. Hình ảnh segment bị gửi lại..... | 26 |

NỘI DUNG CHI TIẾT BÀI THỰC HÀNH

1. Task 1: Phân tích hoạt động giao thức UDP

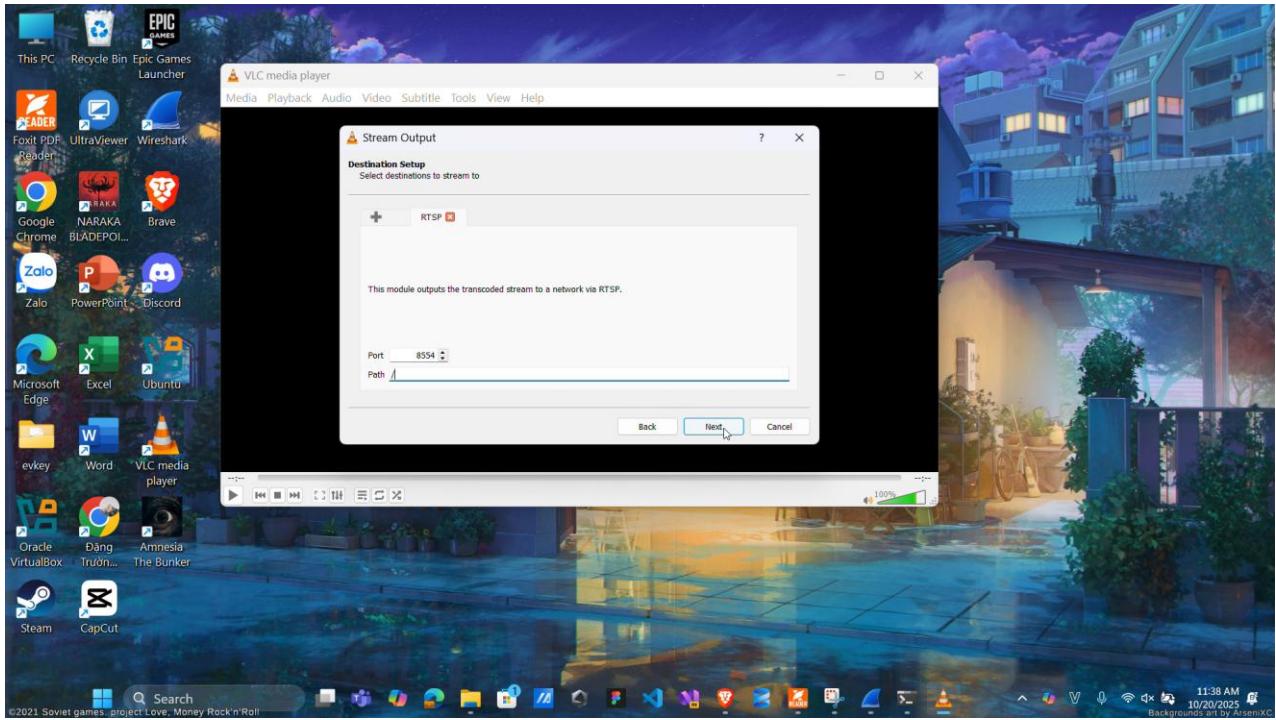
1.1 Streaming Video sử dụng UDP

- Bước 1:** Cài đặt phần mềm VLC media player mới nhất trên máy tính và chuẩn bị 1 video để streaming để máy khác trong cùng mạng nội bộ có thể xem.
- Bước 2:** Stream một video từ máy tính bằng VLC (đóng vai trò như server) sử dụng giao thức RTSP (Real Time Streaming Protocol) bằng các bước sau:
 - Chọn menu Media » Stream » Add » Chọn video cần stream



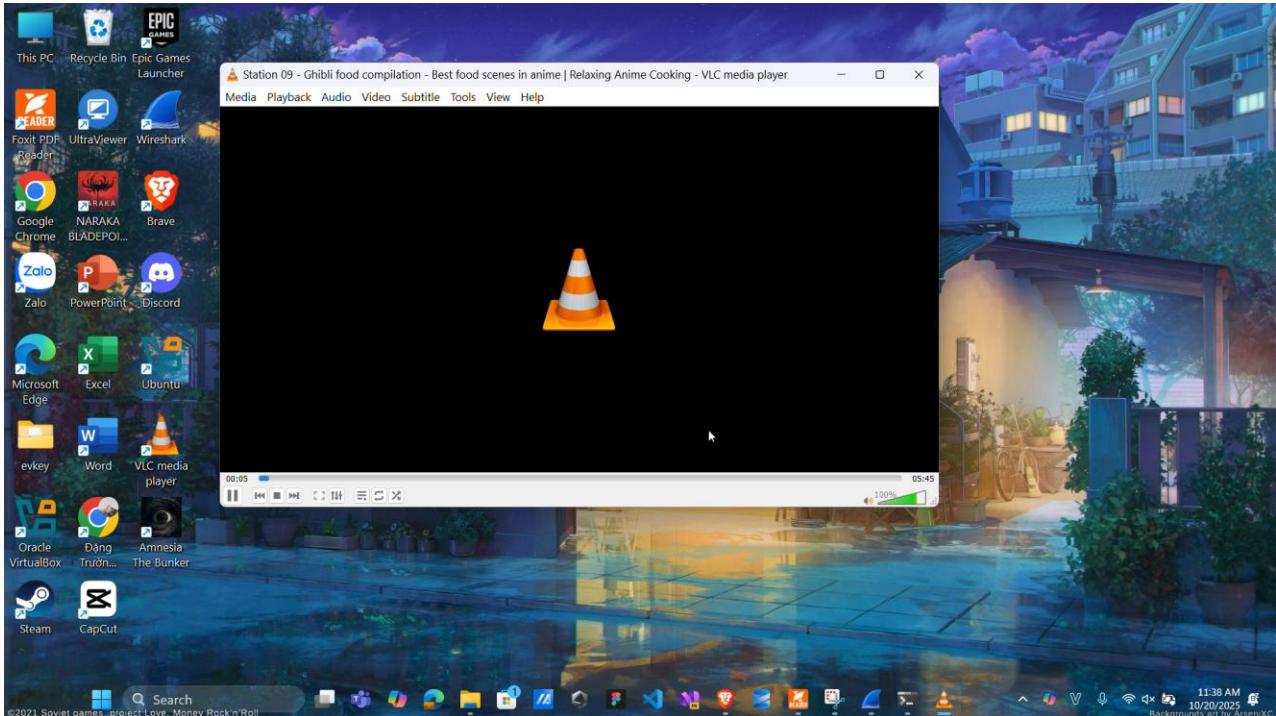
Hình 1. Chọn video cần stream

- Chọn Stream » Next, chọn RTSP trong phần New Destination và chọn Add



Hình 2. Chọn RTSP và Port mặc định khi stream là 8554

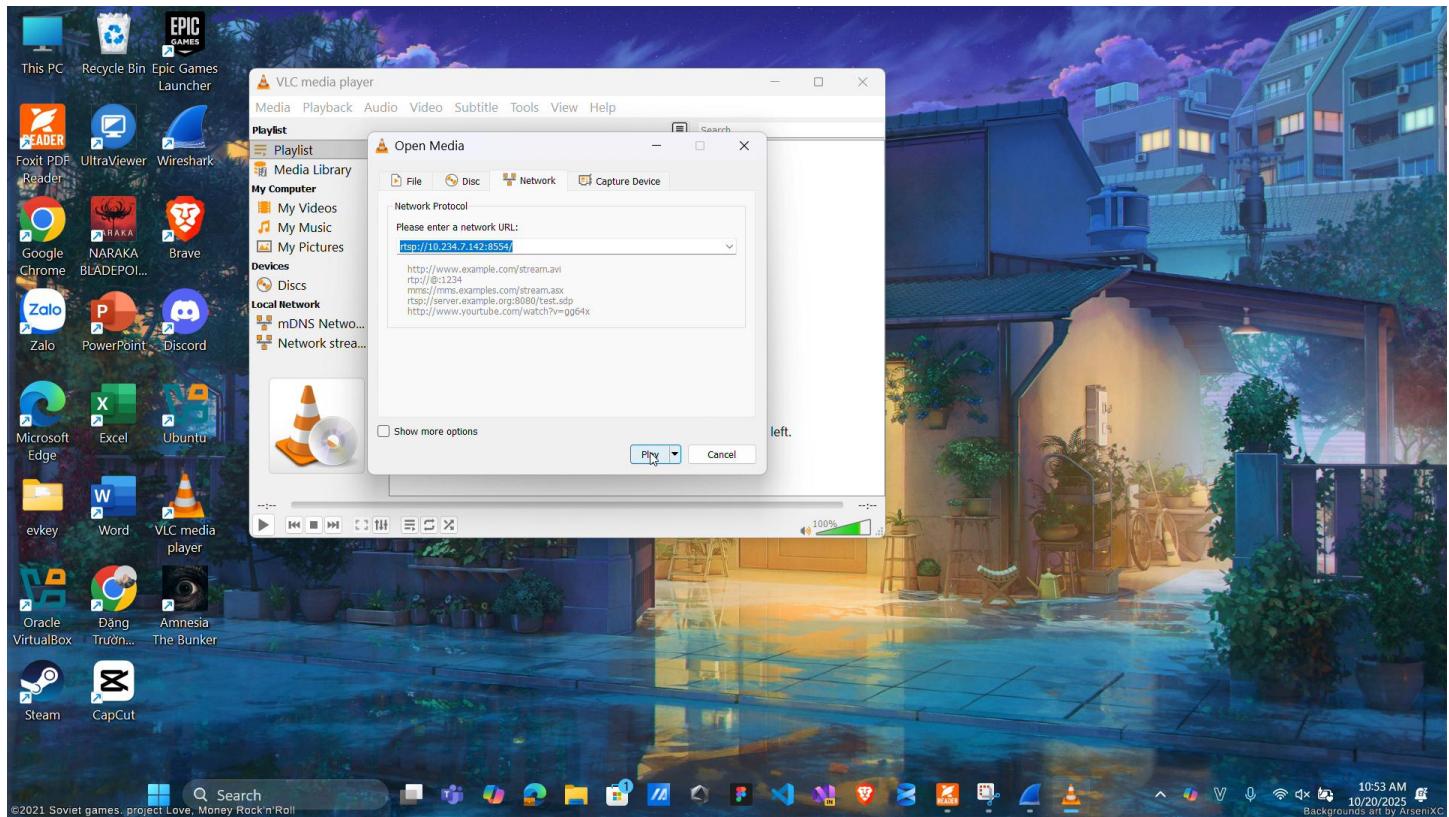
- Giữ nguyên các tùy chọn mặc định ở các phần tiếp theo và chọn **Stream** để bắt đầu truyền tải video



Hình 3. Bắt đầu stream

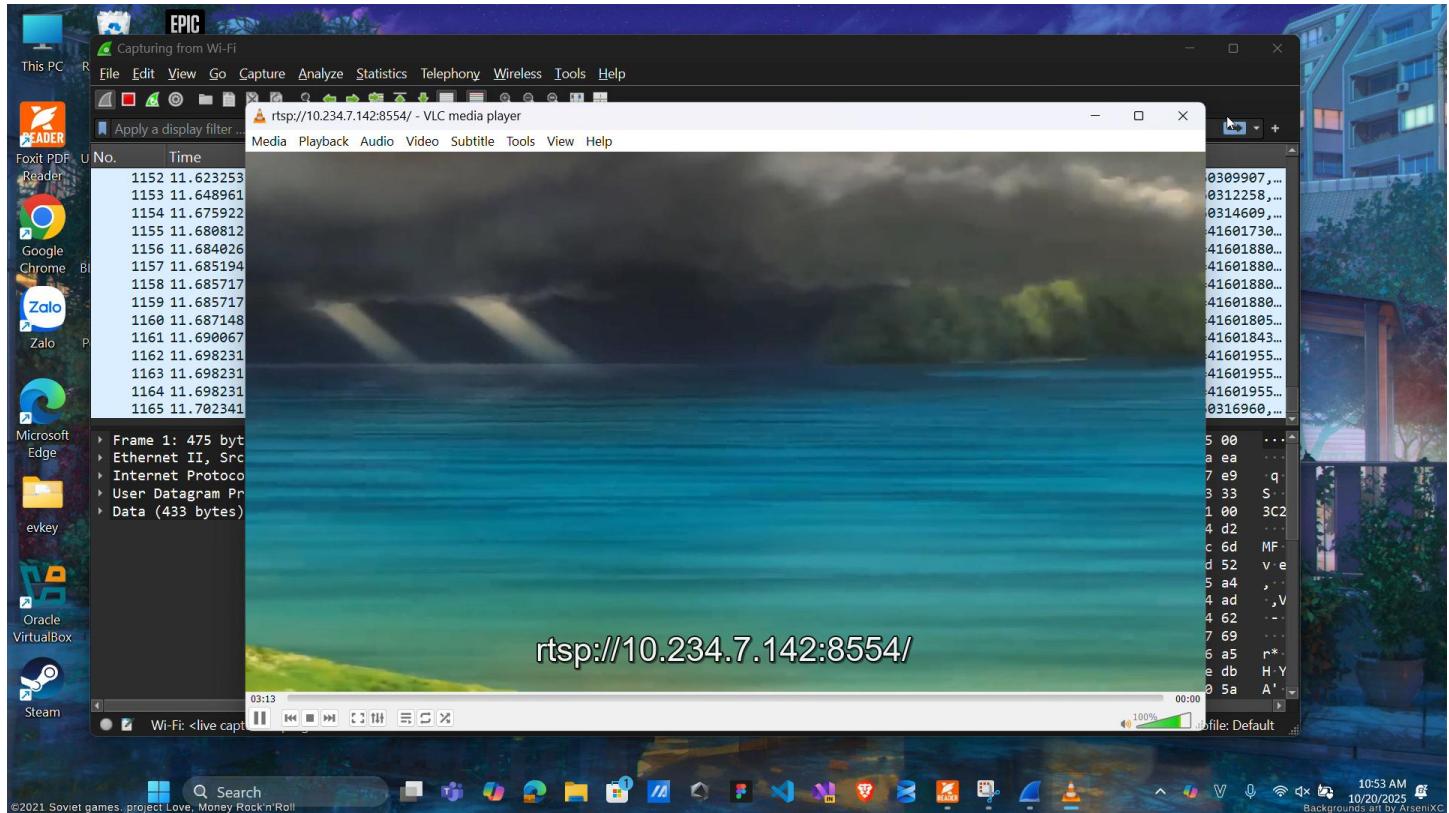
1.2 Tiến hành bắt gói tin UDP khi streaming video

- **Bước 3:** Mở Wireshark và bắt đầu bắt gói tin
- **Bước 4:** Truy cập để xem video từ 1 máy server của sinh viên khác bằng cách mở phần mềm VLC » Chọn **Open network stream**
Nhập địa chỉ theo định dạng rtsp://[địa chỉ IP server]:8554/



Hình 4. Truy cập để xem video stream từ máy của sinh viên khác

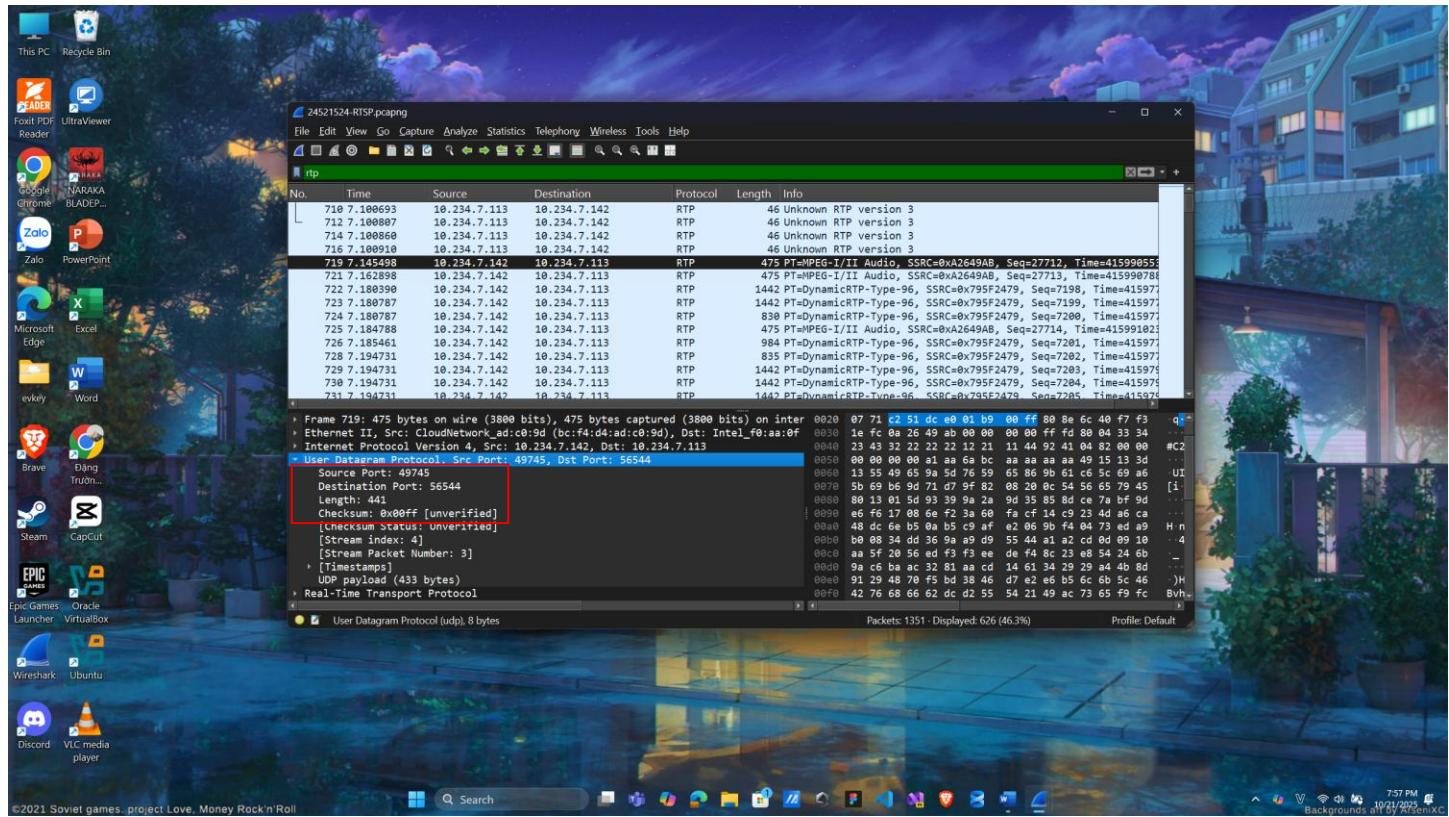
- **Bước 5:** Xem Video một thời gian ngắn



Hình 5. Xem video 1 thời gian ngắn

1.3 Phân tích hoạt động giao thức UDP

- Chọn một gói tin UDP, xác định các trường (field) có trong UDP header và giải thích ý nghĩa của mỗi trường đó?



Hình 6. Các trường có trong UDP header.

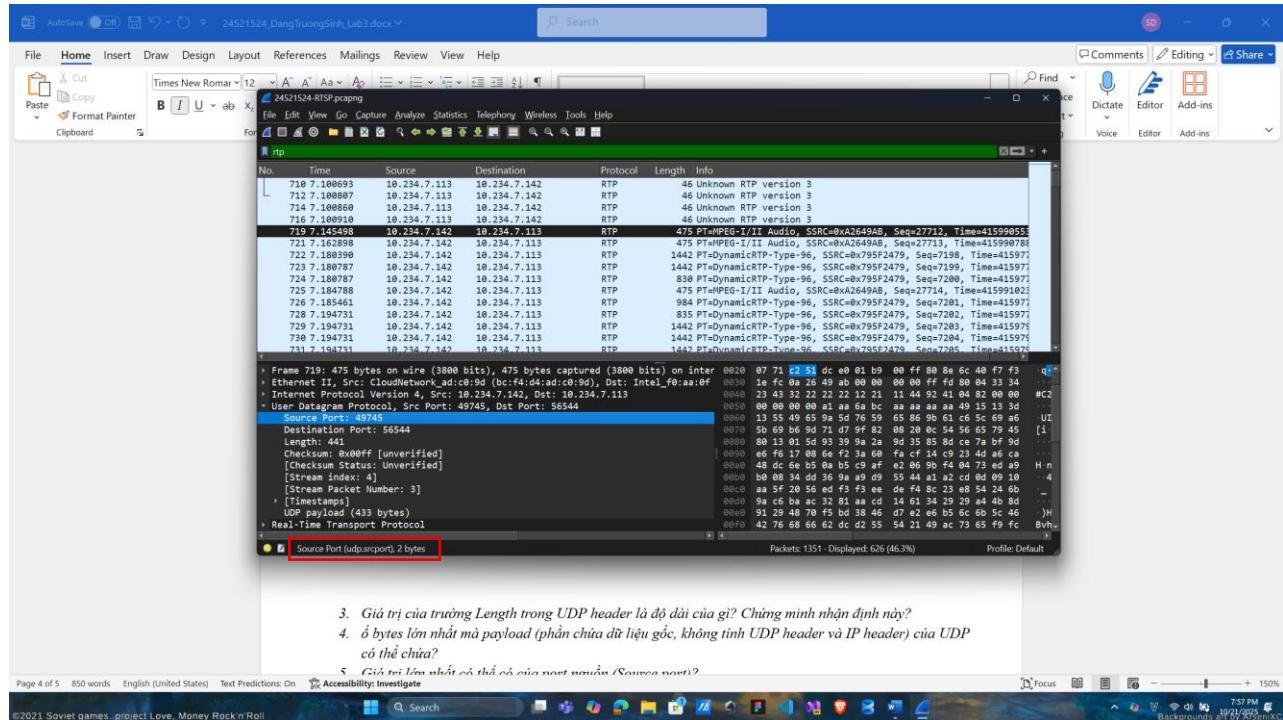
Các trường (field) có trong UDP header

- Source Port:** Địa chỉ của công gửi thông tin.
- Destination Port:** Địa chỉ của công nhận thông tin.
- Length:** Tổng độ dài (bytes) của gói tin
- Checksum:** Mã kiểm tra lỗi giúp phát hiện dữ liệu bị hỏng trong quá trình truyền.

- Qua thông tin hiển thị của Wireshark, xác định độ dài (tính theo byte) của mỗi trường trong UDP header?

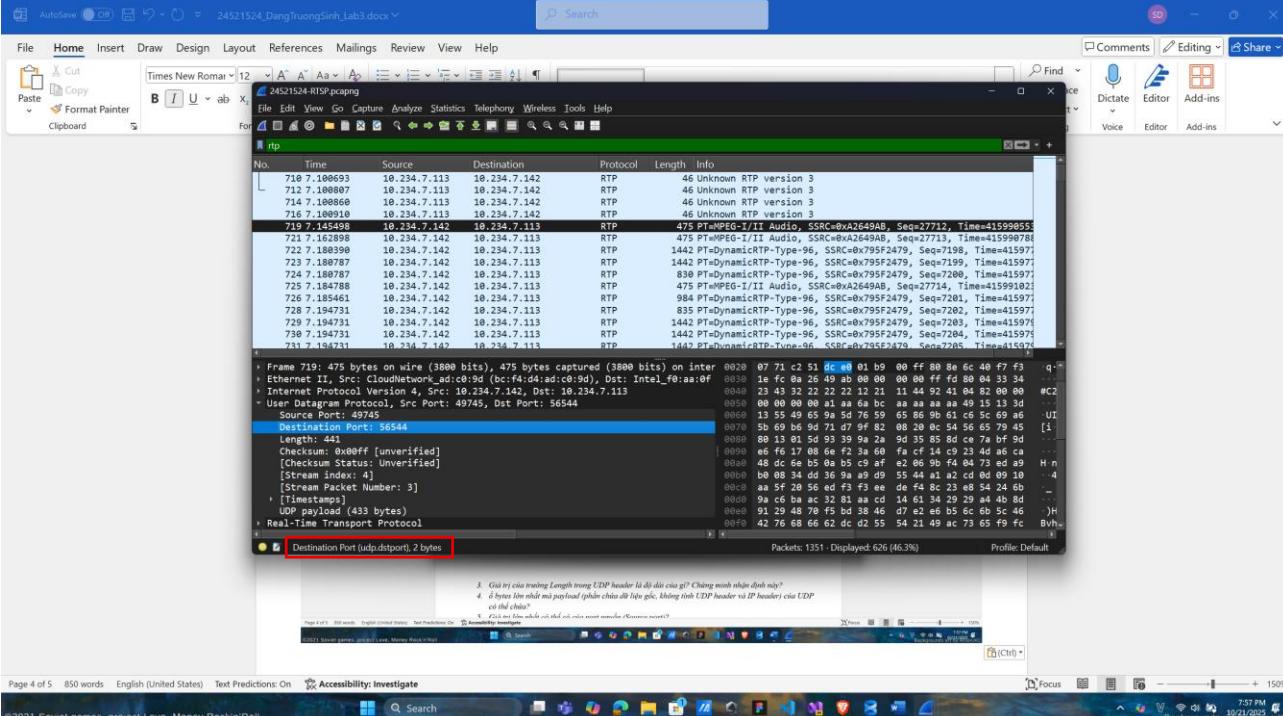
Độ dài của mỗi trường trong UDP header:

- Source Port:** 2 bytes.
- Destination Port:** 2 bytes.
- Length:** 2 bytes.
- Checksum:** 2 bytes.

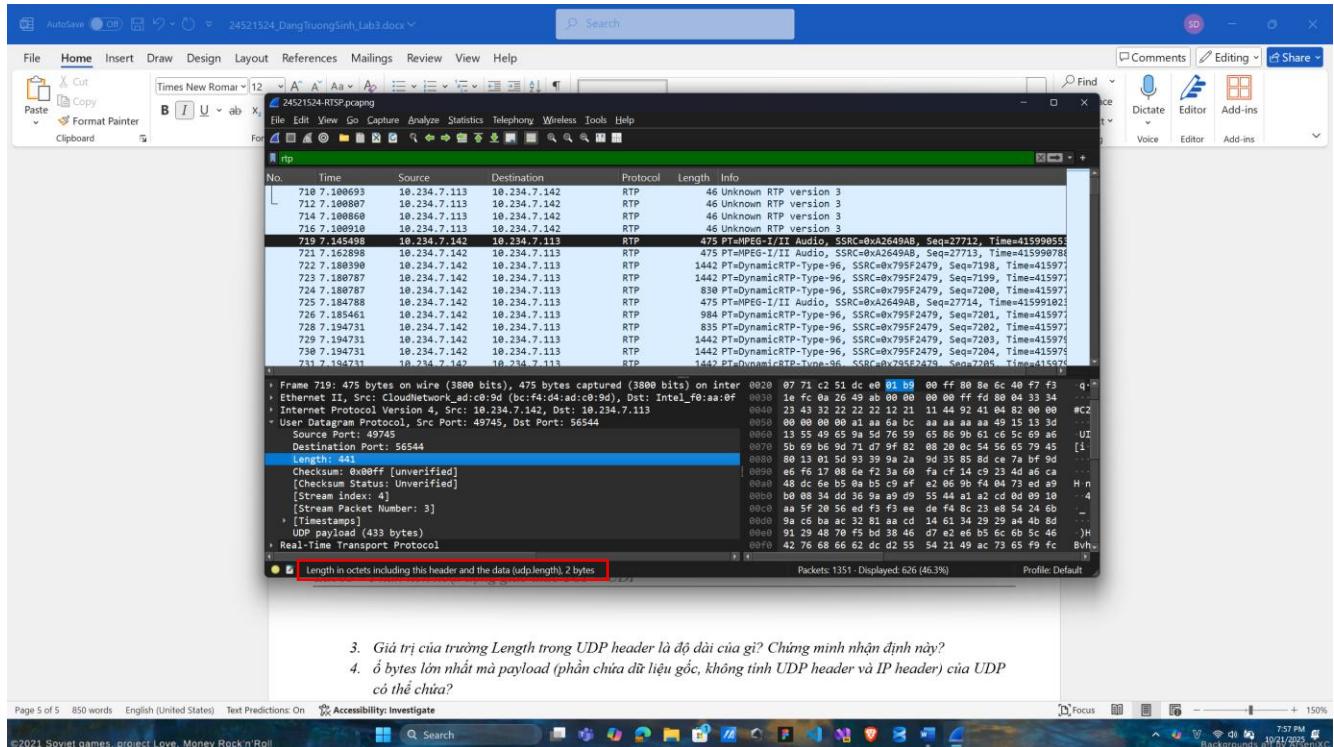


3. Giá trị của trường Length trong UDP header là độ dài của gì? Chứng minh nhận định này?
4. Ở bytes lớn nhất mà payload (phần chứa dữ liệu gốc, không tính UDP header và IP header) của UDP có thể chứa?

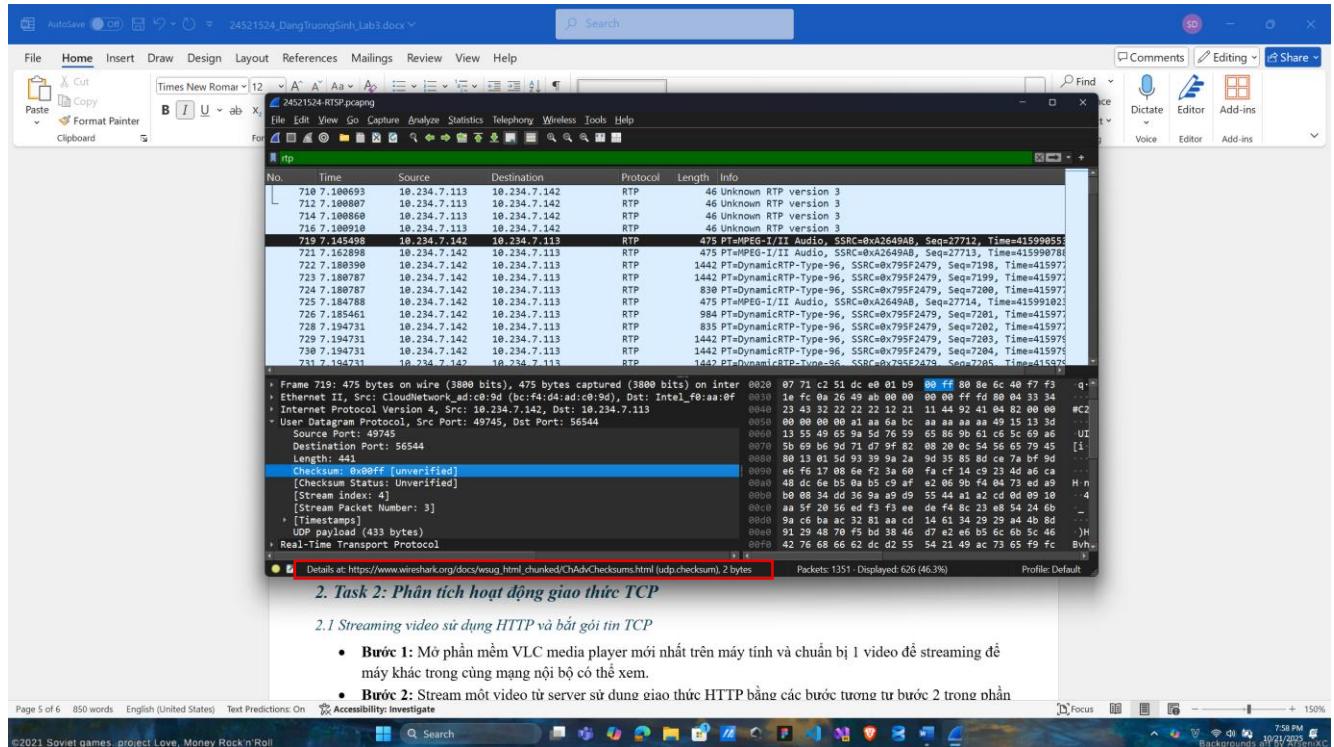
5. Giá trị lớn nhất có thể có của port nguồn (Source port)?



Hình 8. Độ dài của trường Destination Port



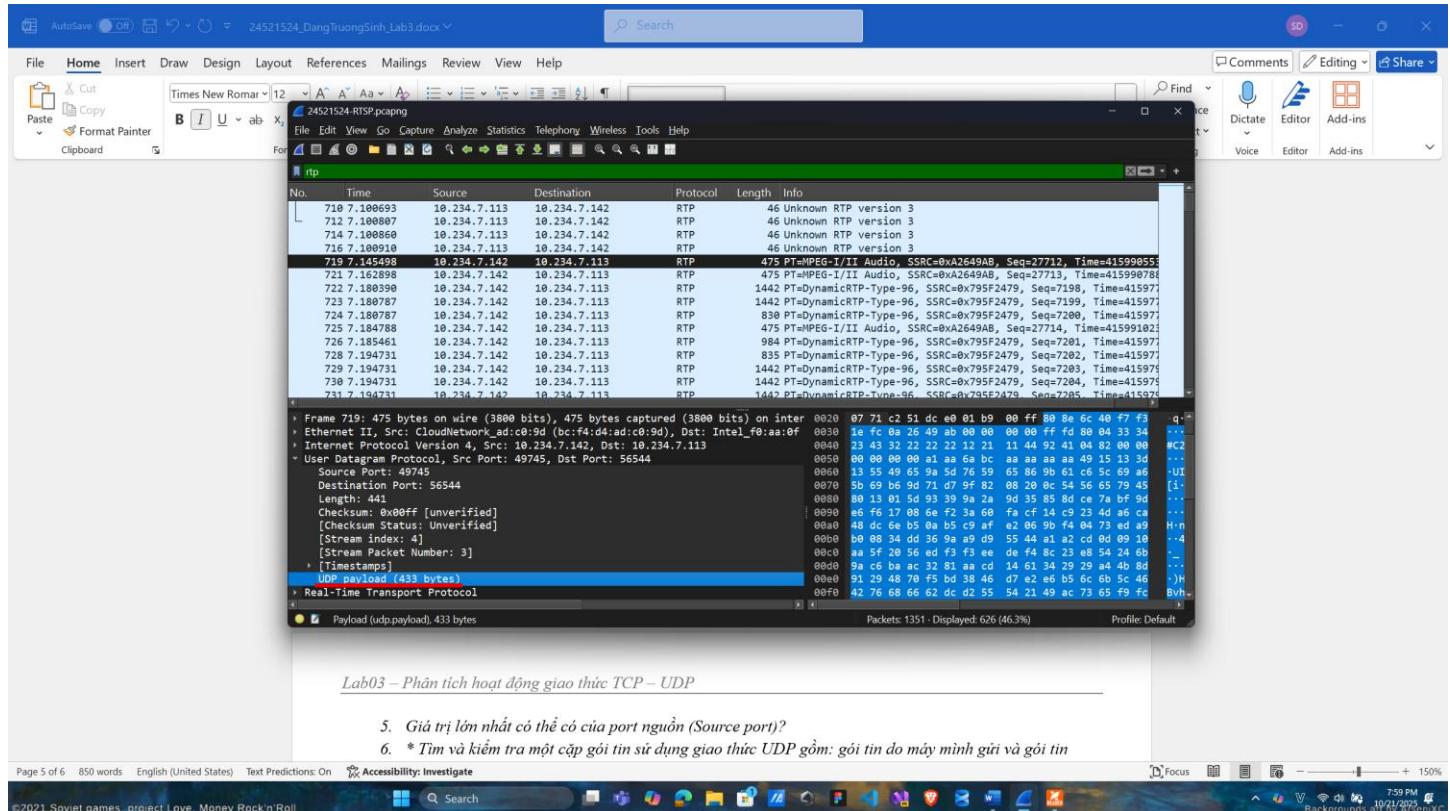
Hình 9. Độ dài của trường Length



Hình 10. Độ dài của trường Checksum

3. Giá trị của trường **Length** trong **UDP header** là độ dài của gì? Chứng minh nhận định này?

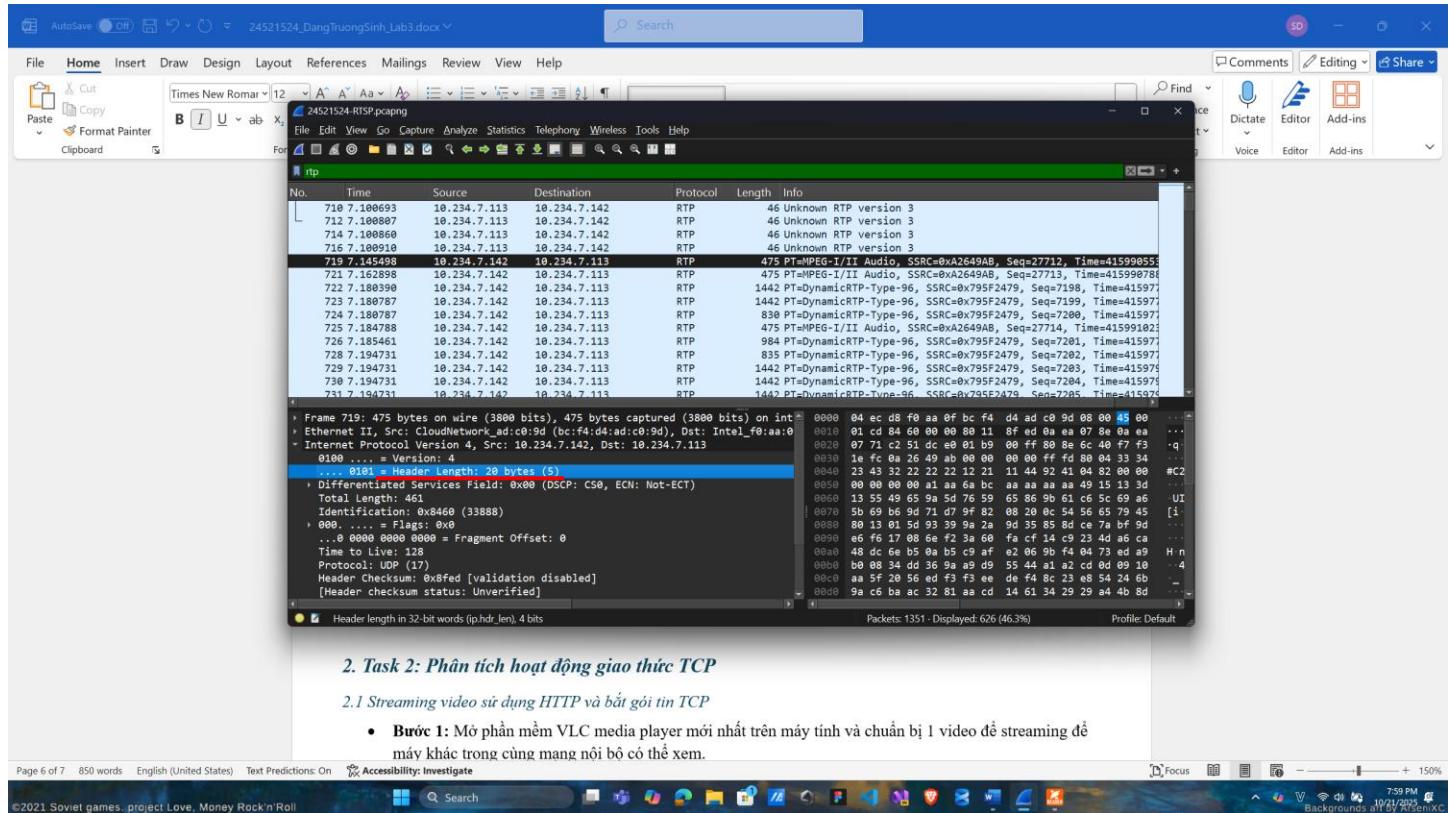
- Giá trị trường **Length** trong **UDP header** là tổng độ dài của các trường trong **UDP header** và **UDP payload**.
- Ở các trường trong **UDP header**, như câu trên ta thấy, mỗi trường đều có độ dài là **2 bytes**. Do đó tổng độ dài của **UDP header** là **8 bytes**. Ngoài ra độ dài của **UDP payload** là **433 bytes**. Do đó ta có thể thấy rằng trường **Length** trong **UDP header** là $433 + 8 = 441$ bytes.



Hình 11. Độ dài của UDP payload

4. Số bytes lớn nhất mà payload (phần chứa dữ liệu gốc, không tính UDP header và IP header) của UDP có thể chứa?

- Trường **Length** của UDP dài **16 bit** → tối đa **65535 bytes**.
 - IP header = **20 bytes**, UDP header = **8 bytes**.
- Số bytes lớn nhất mà payload có thể chứa là: **65535 - 20 - 8 = 65507 bytes**.

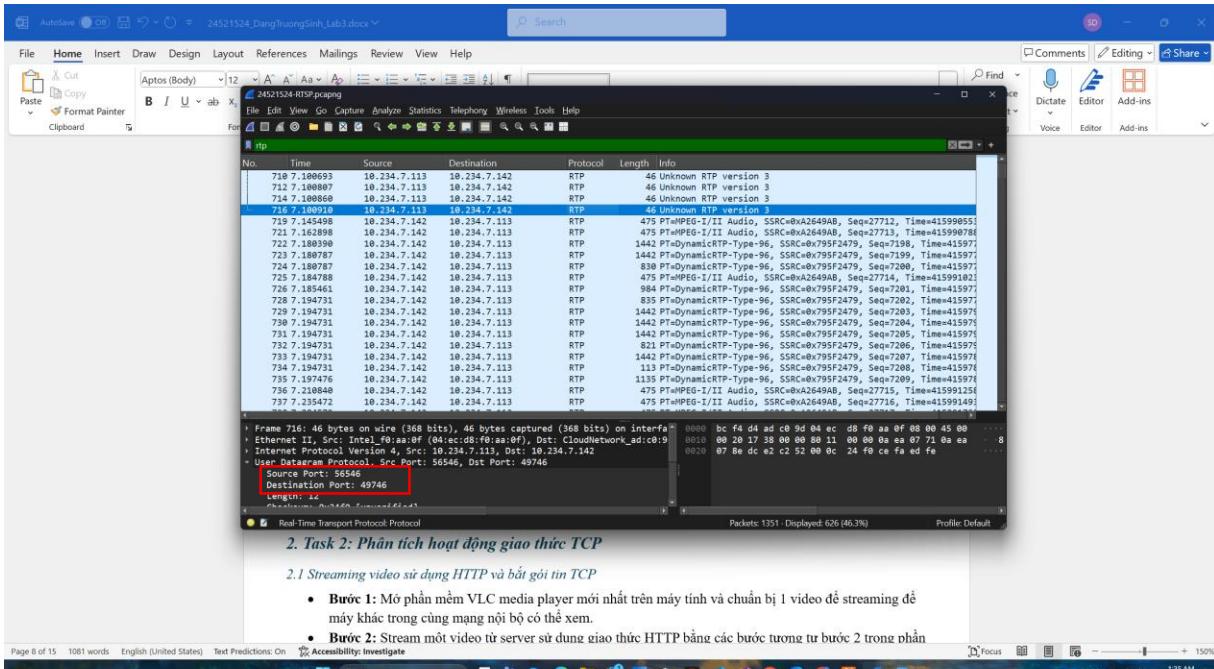


Hình 12. Độ dài của IP header

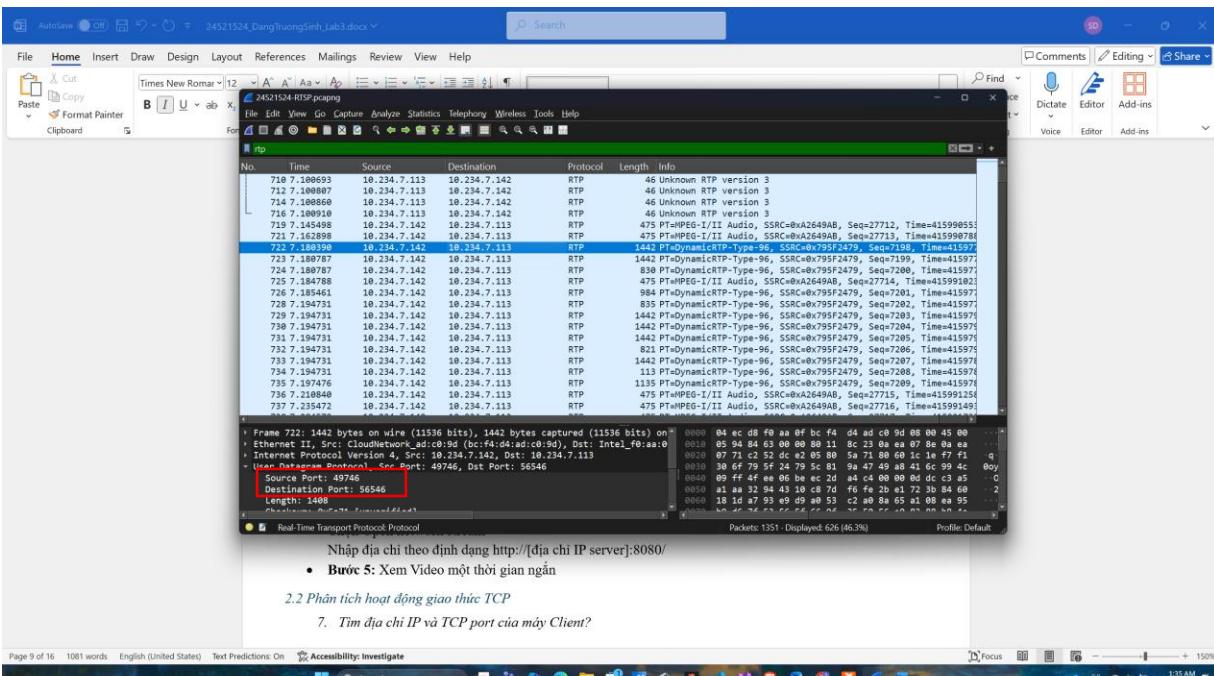
5. Giá trị lớn nhất có thể có của port nguồn (Source port)?

- Source Port** có độ dài là **2 bytes**. Do đó, giá trị lớn nhất có thể có của nó là $2^{16} - 1 = 65535$ bytes.
- * Tìm và kiểm tra một cặp gói tin sử dụng giao thức UDP gồm: gói tin do máy mình gửi và gói tin phản hồi của gói tin đó. Miêu tả mối quan hệ về port number của 2 gói tin này
 - Ở gói tin máy mình gửi (gói 716):
 - Source Port:** 56546
 - Destination Port:** 49746
 - Ở gói tin phản hồi (gói 722):
 - Source Port:** 49746
 - Destination Port:** 56546

→ Trong giao thức UDP, gói phản hồi có **Source port** và **Destination port** đảo ngược so với gói gửi đi. Port dịch vụ (server) thường cố định, còn port phía client là port tạm.



Hình 13. Gói tin máy mình gửi

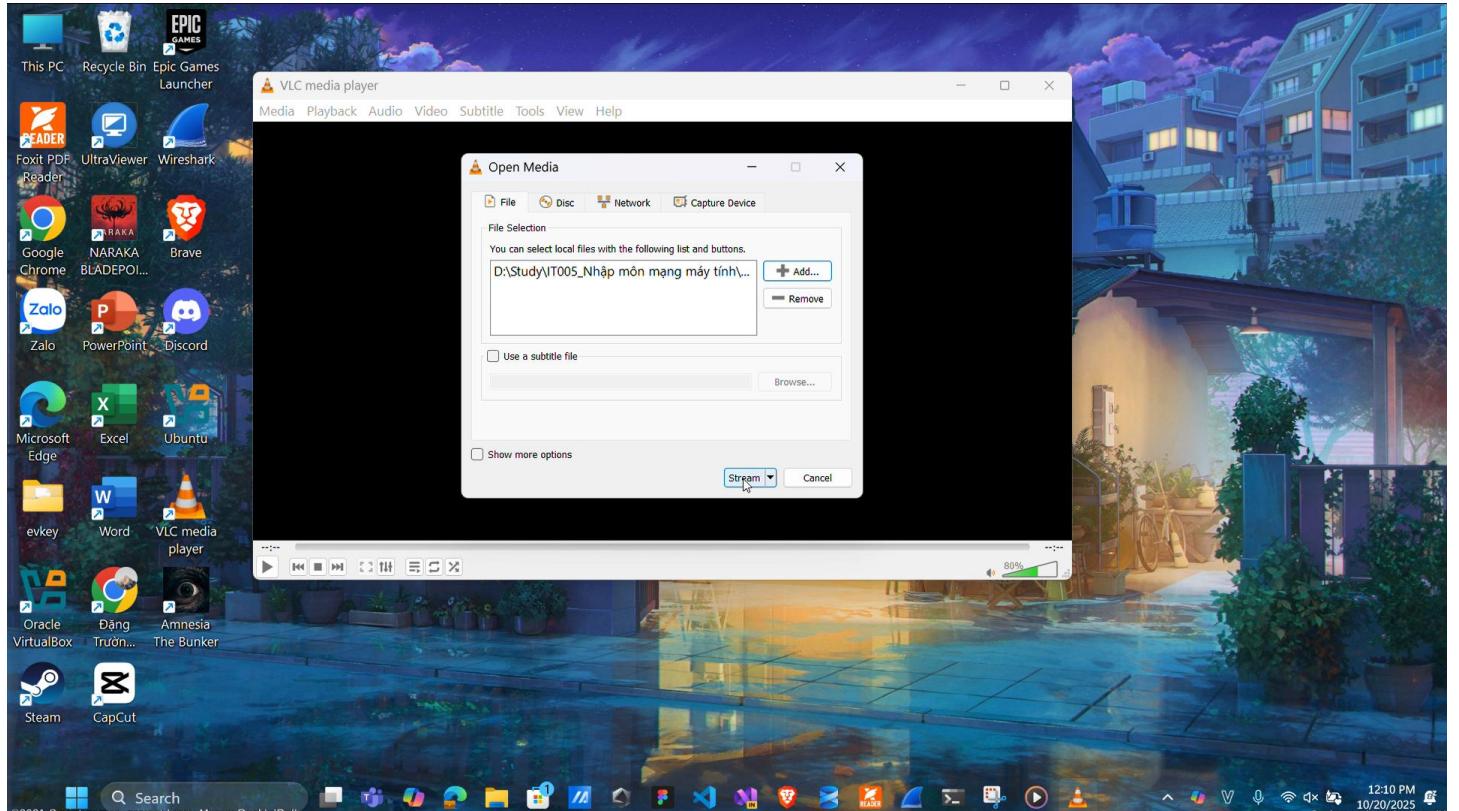


Hình 14. Gói tin phản hồi

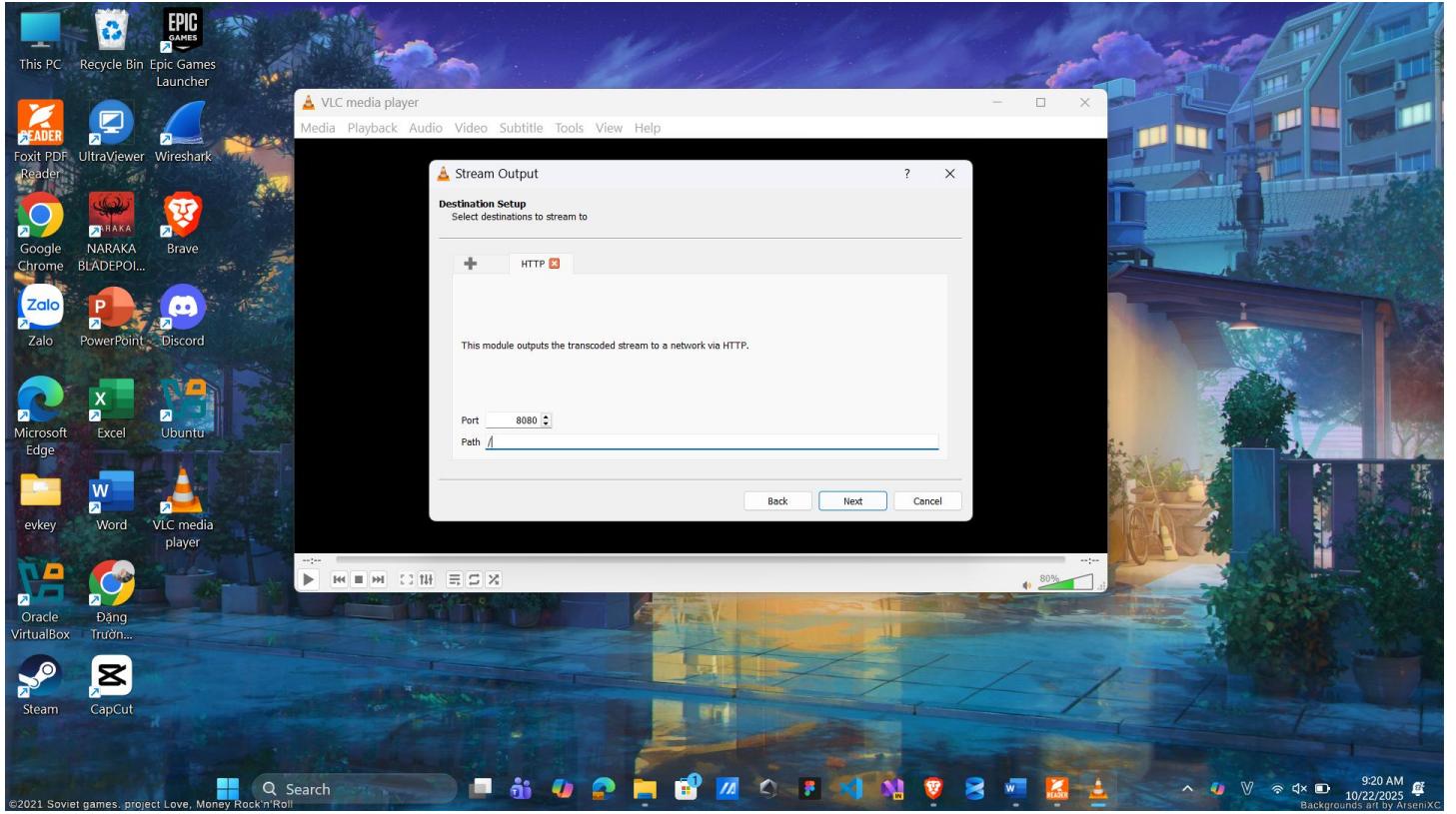
2. Task 2: Phân tích hoạt động giao thức TCP

2.1 Streaming video sử dụng HTTP và bắt gói tin TCP

- **Bước 1:** Mở phần mềm VLC media player mới nhất trên máy tính và chuẩn bị 1 video để streaming để máy khác trong cùng mạng nội bộ có thể xem.
- **Bước 2:** Stream một video từ server sử dụng giao thức HTTP bằng các bước tương tự bước 2 trong phần UDP. Tuy nhiên, trong phần New Destination, thay vì chọn RTSP, chọn HTTP với port mặc định là 8080

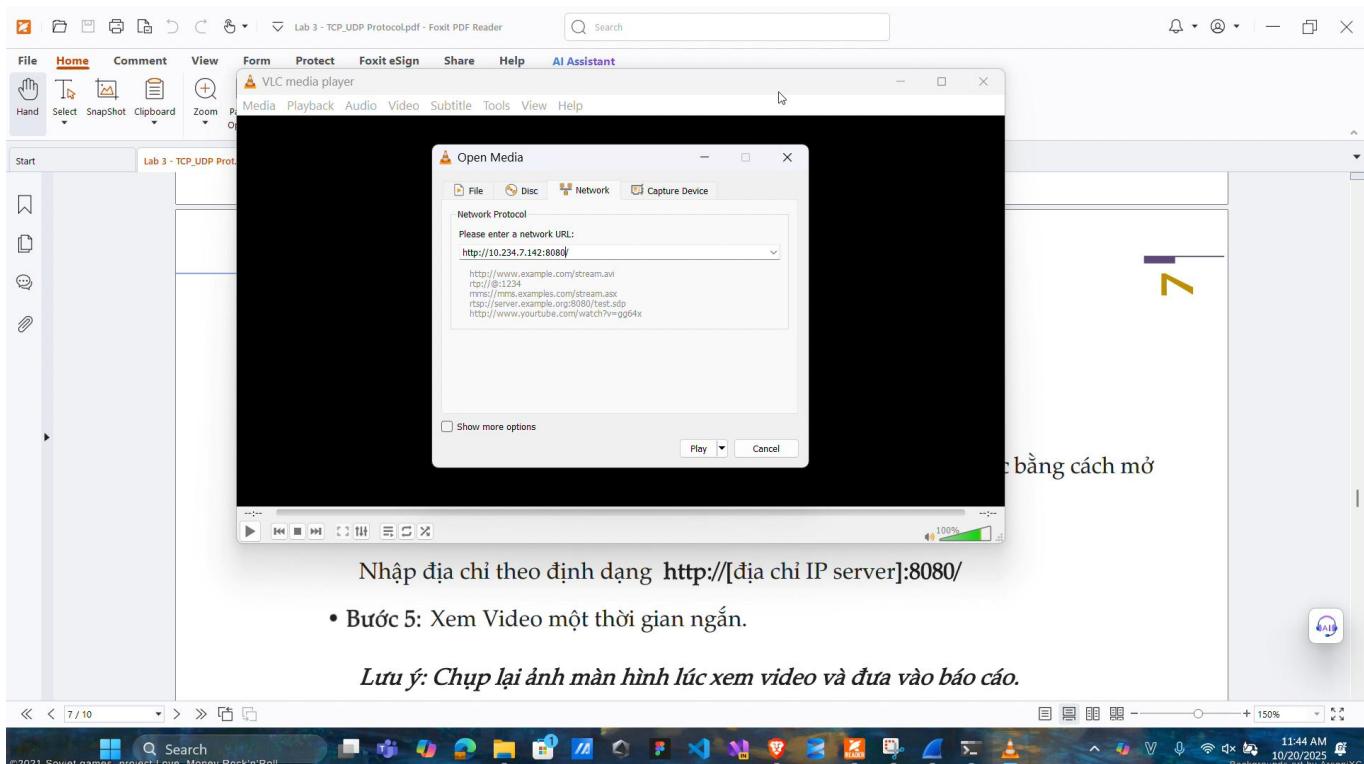


Hình 15. Chọn video để stream



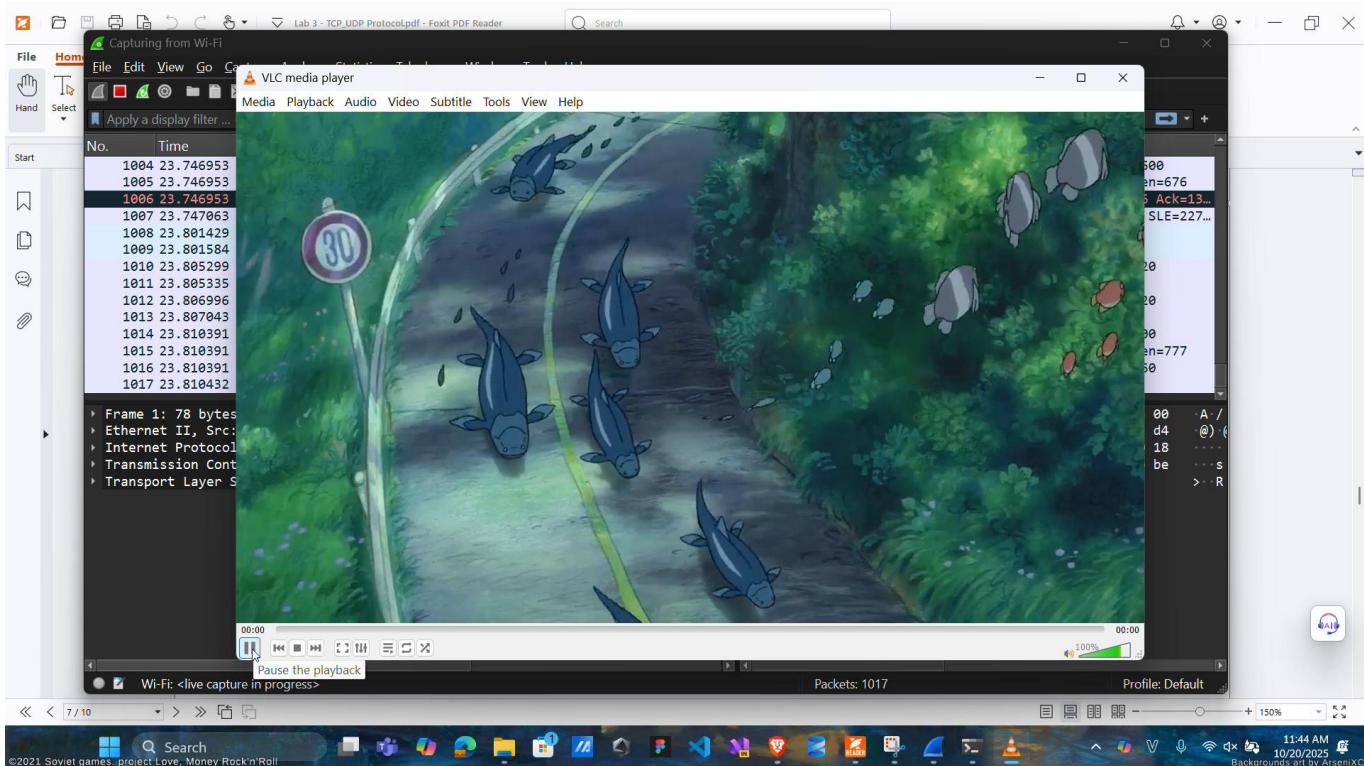
Hình 16. Chọn HTTP với Port mặc định của là 8080

- **Bước 3:** Tiến hành bắt gói tin TCP khi streaming video:
Mở Wireshark và bắt đầu bắt gói tin.
- **Bước 4:** Truy cập để xem video từ 1 máy server của sinh viên khác bằng cách mở phần mềm VLC »
Chọn **Open network stream**
Nhập địa chỉ theo định dạng http://[địa chỉ IP server]:8080/



Hình 17. Truy cập để xem video stream của sinh viên khác

- **Bước 5:** Xem Video một thời gian ngắn

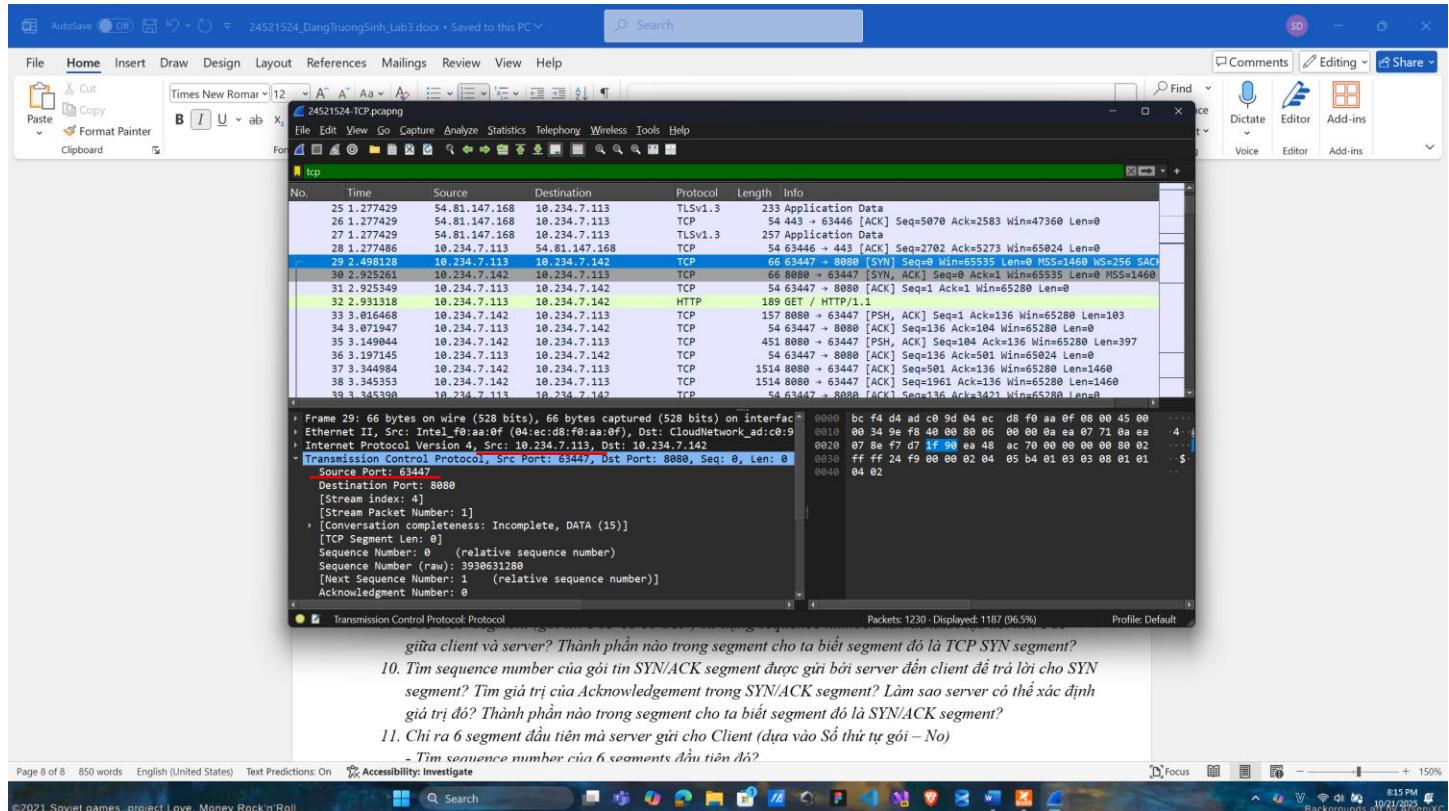


Hình 18. Xem video 1 thời gian ngắn

2.2 Phân tích hoạt động giao thức TCP

7. Tìm địa chỉ IP và TCP port của máy Client?

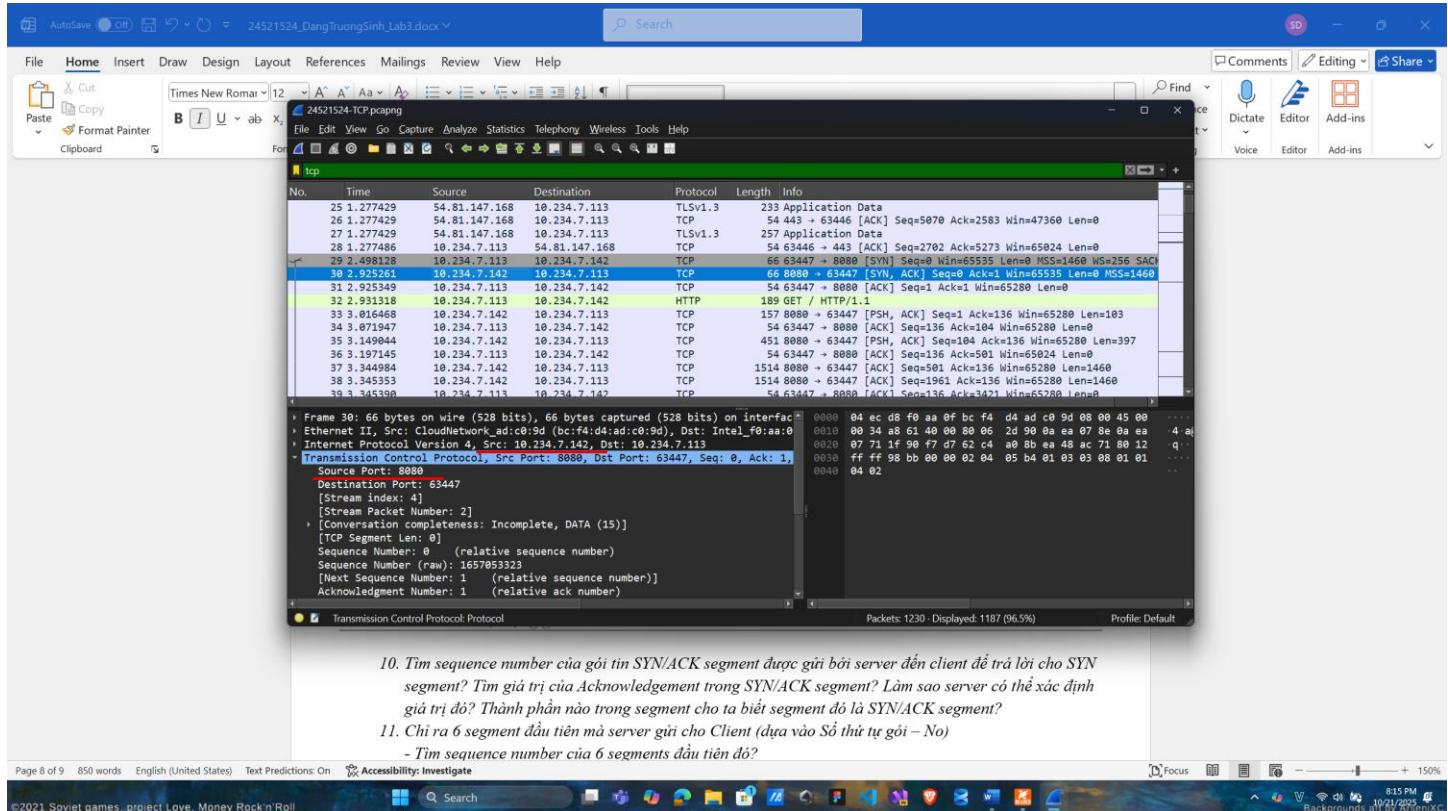
- Địa chỉ IP máy Client: 10.234.7.113
- TCP Port: 63447



Hình 19. Địa chỉ IP và TCP Port của Client

8. Tìm địa chỉ IP của Server? Kết nối TCP dùng để gửi và nhận các segments sử dụng port nào?

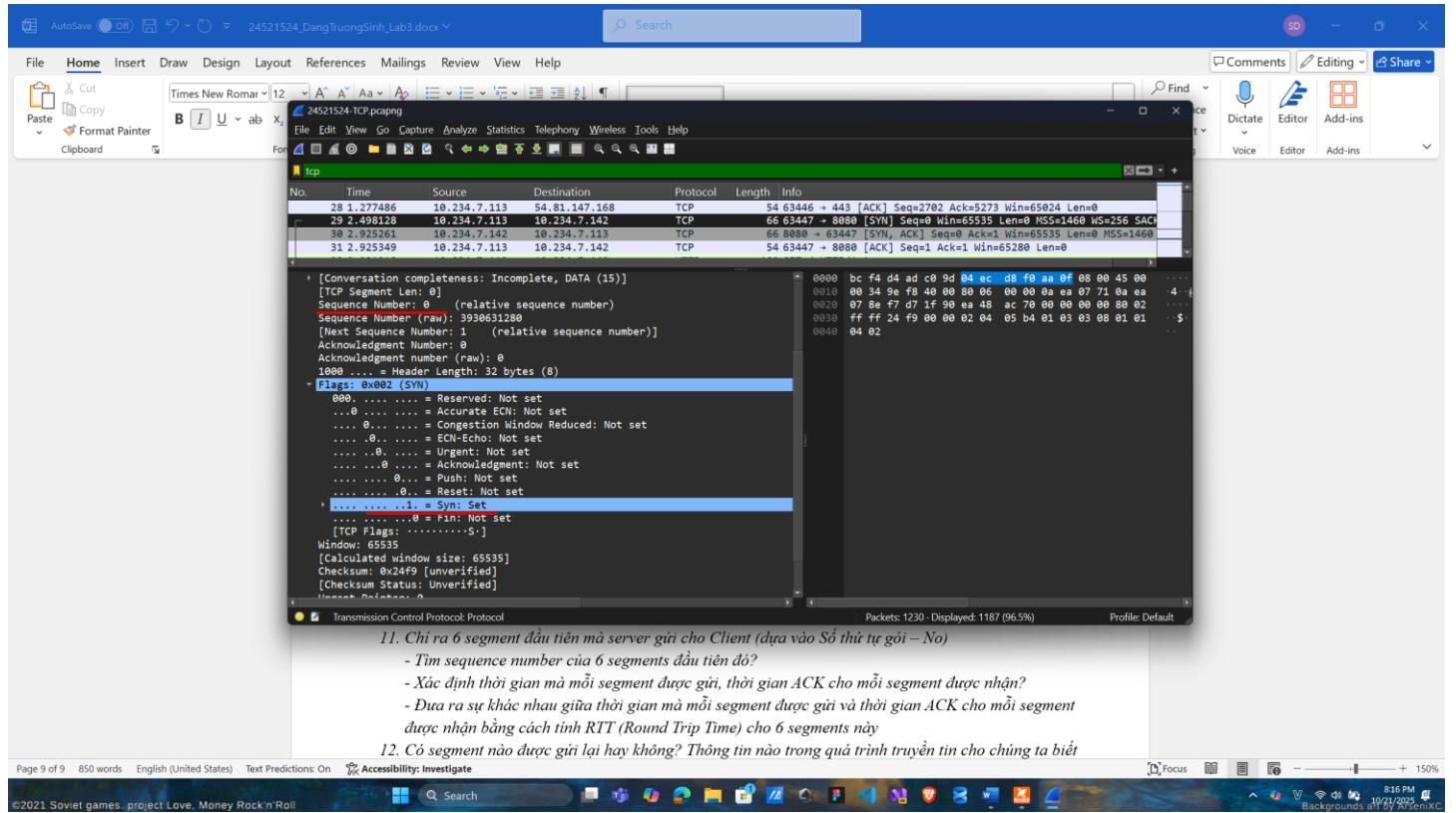
- Địa chỉ IP Sever: 10.234.7.142
- Port: 8080



Hình 20. Địa chỉ IP và TCP Port của Sever

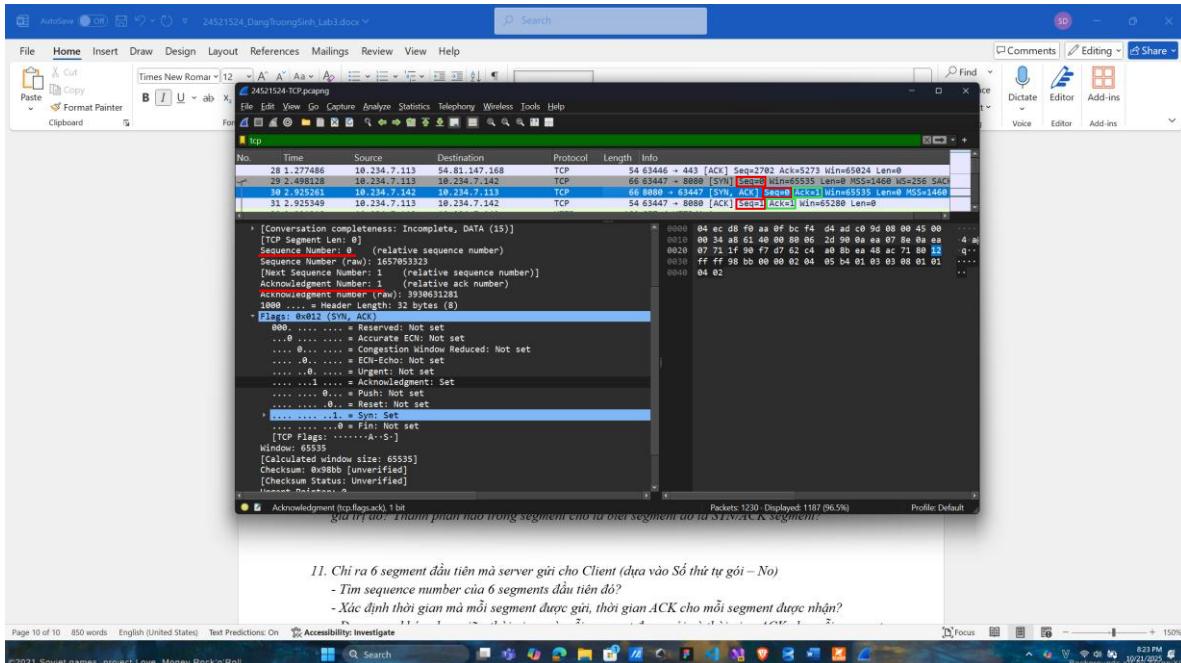
9. TCP SYN segment (gói tin TCP có cờ SYN) sử dụng sequence number nào để khởi tạo kết nối TCP giữa client và server? Thành phần nào trong segment cho ta biết segment đó là TCP SYN segment?

- TCP SYN segment (gói tin TCP có cờ SYN) sử dụng sequence number là 0 để khởi tạo kết nối TCP.
- Ta thấy Flags (SYN) được set là 1 → TCP SYN segment.



Hình 21. Gói tin TCP có cờ SYN

10. Tìm sequence number của gói tin SYN/ACK segment được gửi bởi server đến client để trả lời cho SYN segment? Tìm giá trị của Acknowledgement trong SYN/ACK segment? Làm sao server có thể xác định giá trị đó? Thành phần nào trong segment cho ta biết segment đó là SYN/ACK segment?

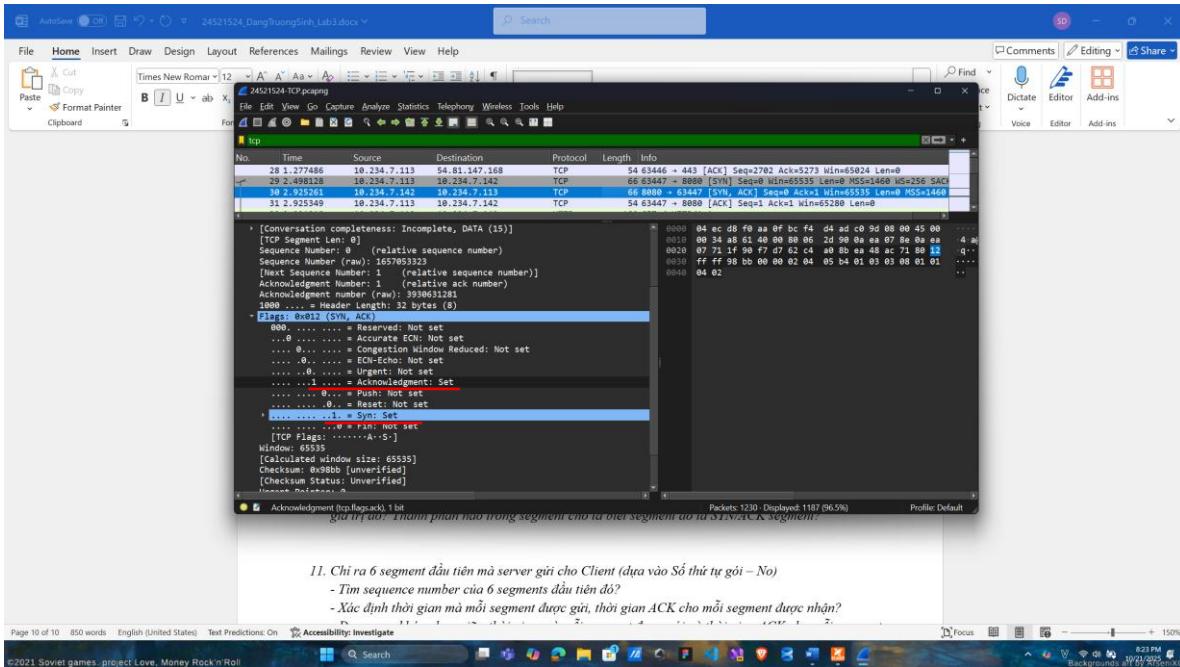


Hình 22. Sequence number và giá trị Acknowledgement trong SYN/ACK segment

- Từ ảnh trên ta có thấy **sequence number** của gói tin **SYN/ACK segment** được gửi bởi **server** đến **client** để trả lời cho **SYN segment** là 0.
- Từ ảnh trên, ta thấy rằng giá trị **Acknowledgement** trong **SYN/ACK segment** là 1.

Cách Sever xác định được giá trị **Acknowledgement**:

- Ban đầu, ở **SYN**. **Client** sẽ khởi tạo giá trị **Sequence number** của **SYN** là 0.
- Sau đó, ở bước thứ 2, giá trị của **Acknowledgement** tinh gói tin **[SYN, ACK]** sẽ được **Server** bằng giá trị **Sequence(client)** + 1. Ở ví dụ này, nó sẽ giá trị **Acknowledgement** = 1.
- Tương tự ở gói tin **[ACK]** nó cũng được tính bằng giá trị **Sequence (sever)** + 1. Tức khi này giá trị **ACK** = 1

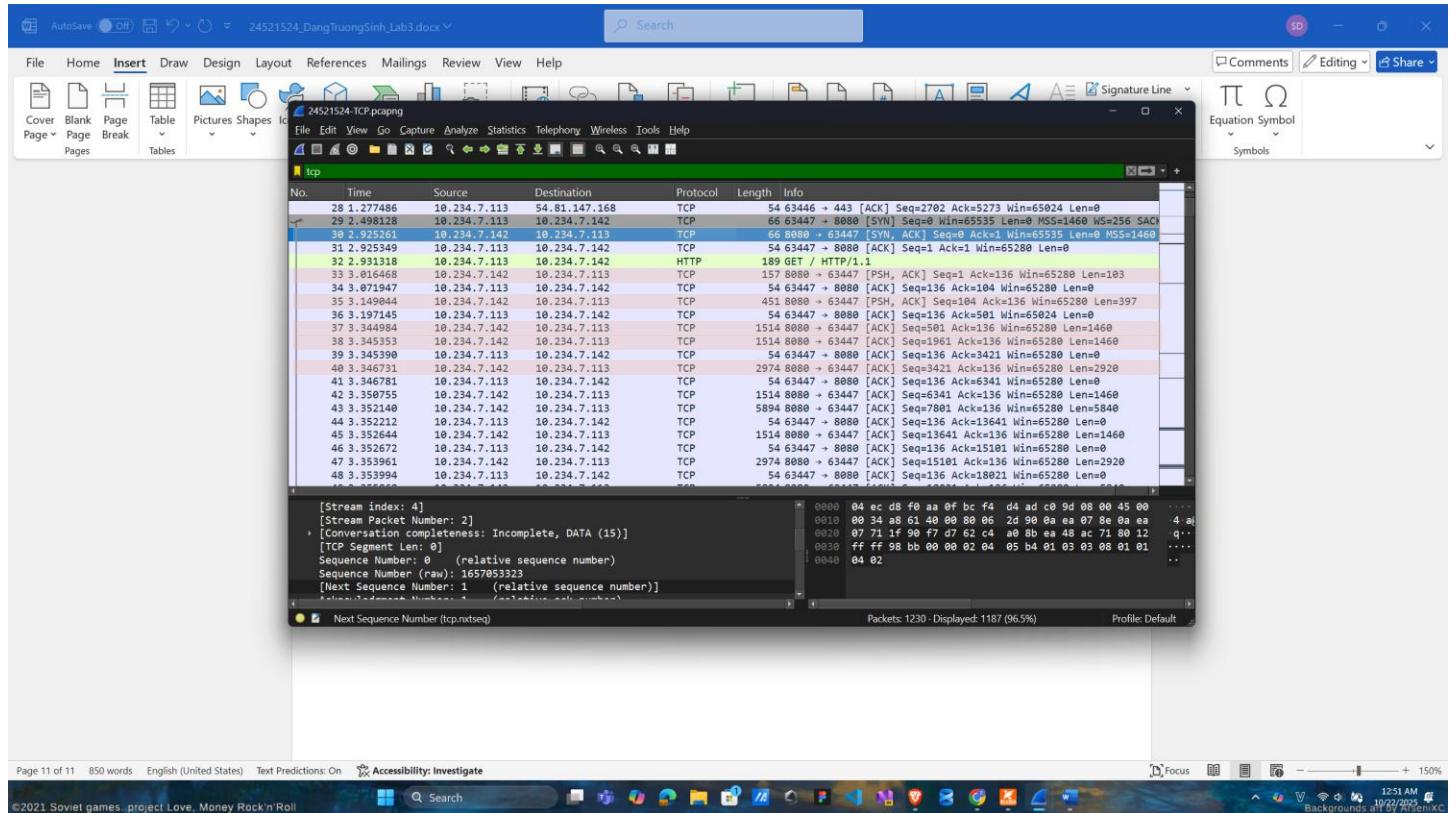


Hình 23. Nhận biết SYN/ACK segment

Ta thấy rằng, trong ảnh ở trường **Flags**, **SYN** và **Acknowledgement** được gán giá trị bằng 1. Do đó, ta có thể nhận biết được rằng đây là **SYN/ACK segment**.

11. Chỉ ra 6 segment đầu tiên mà server gửi cho Client (dựa vào Số thứ tự gói – No)

- Tìm sequence number của 6 segments đầu tiên đó?
- Xác định thời gian mà mỗi segment được gửi, thời gian ACK cho mỗi segment được nhận?
- Đưa ra sự khác nhau giữa thời gian mà mỗi segment được gửi và thời gian ACK cho mỗi segment được nhận bằng cách tính RTT (Round Trip Time) cho 6 segments này



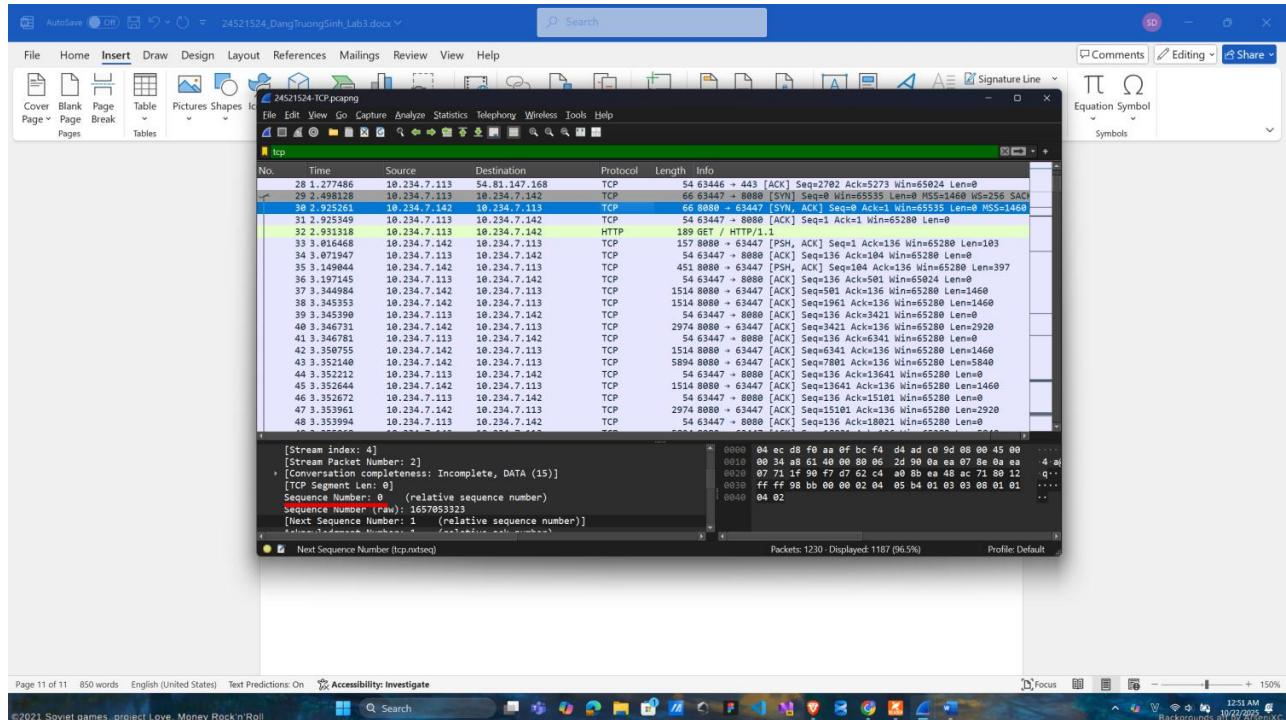
Hình 24. 6 segment đầu tiên mà Sever gửi cho Client

6 Segment đầu tiên mà server gửi cho Client và Sequence Number của 6 segment

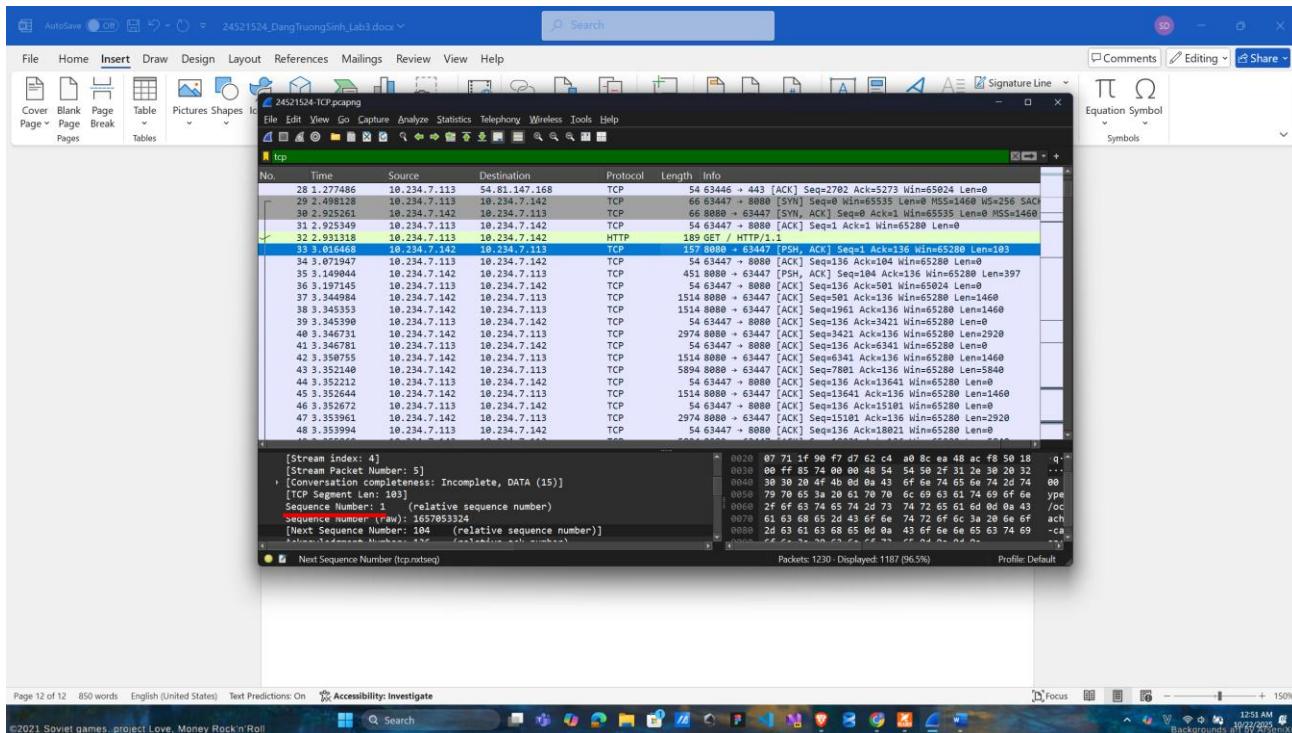
| Segment (No.) | Sequence Number |
|---------------|-----------------|
| 30 | 0 |
| 33 | 1 |
| 35 | 104 |
| 37 | 501 |
| 38 | 1961 |
| 40 | 3421 |

Thời gian mà mỗi segment được gửi, thời gian ACK cho mỗi segment được nhận và RTT

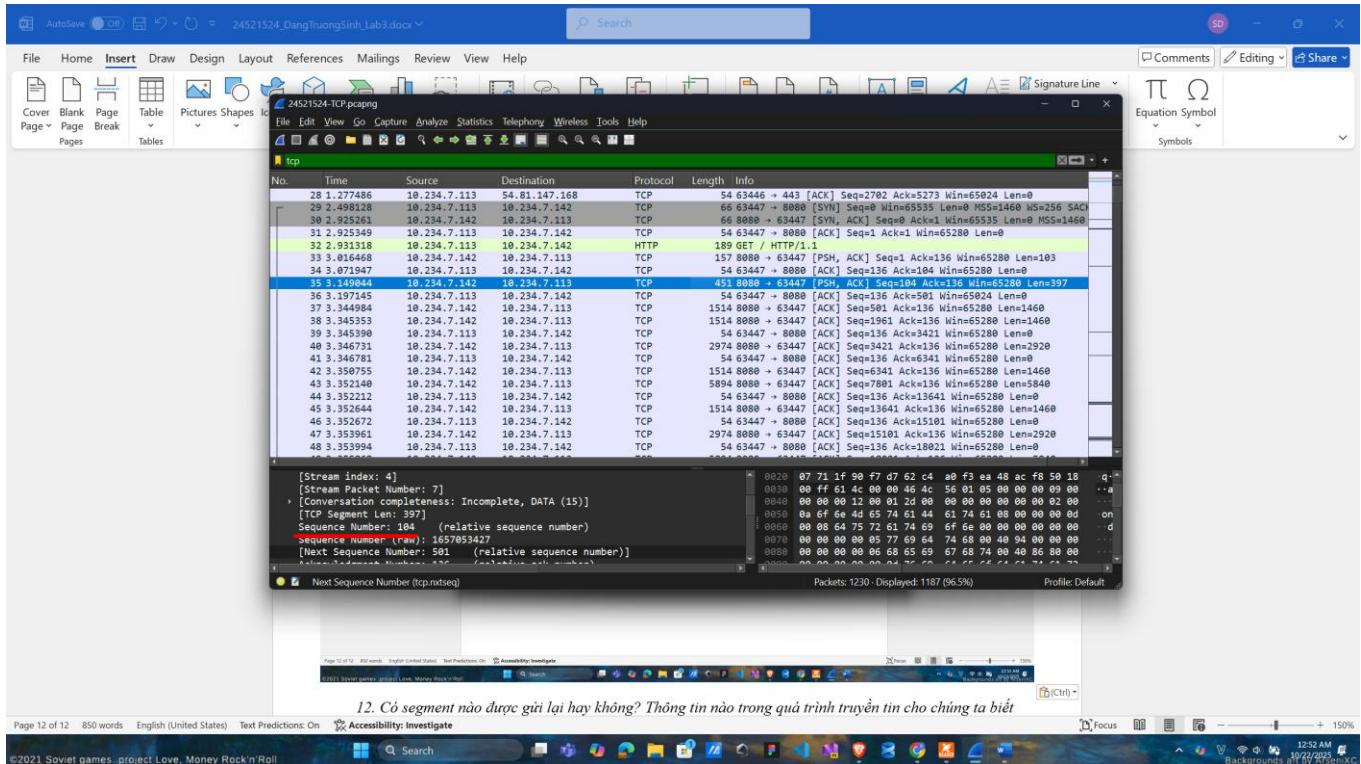
| STT | Thời gian gửi | Thời gian nhận ACK | RTT |
|-----|---------------|--------------------|----------|
| 30 | 2.925261 | 2.925349 | 0.000088 |
| 33 | 3.016468 | 3.071947 | 0.055479 |
| 35 | 3.149044 | 3.197145 | 0.048101 |
| 37 | 3.344984 | 3.345390 | 0.000406 |
| 38 | 3.345353 | 3.345390 | 0.000037 |
| 40 | 3.346731 | 3.346781 | 0.000050 |



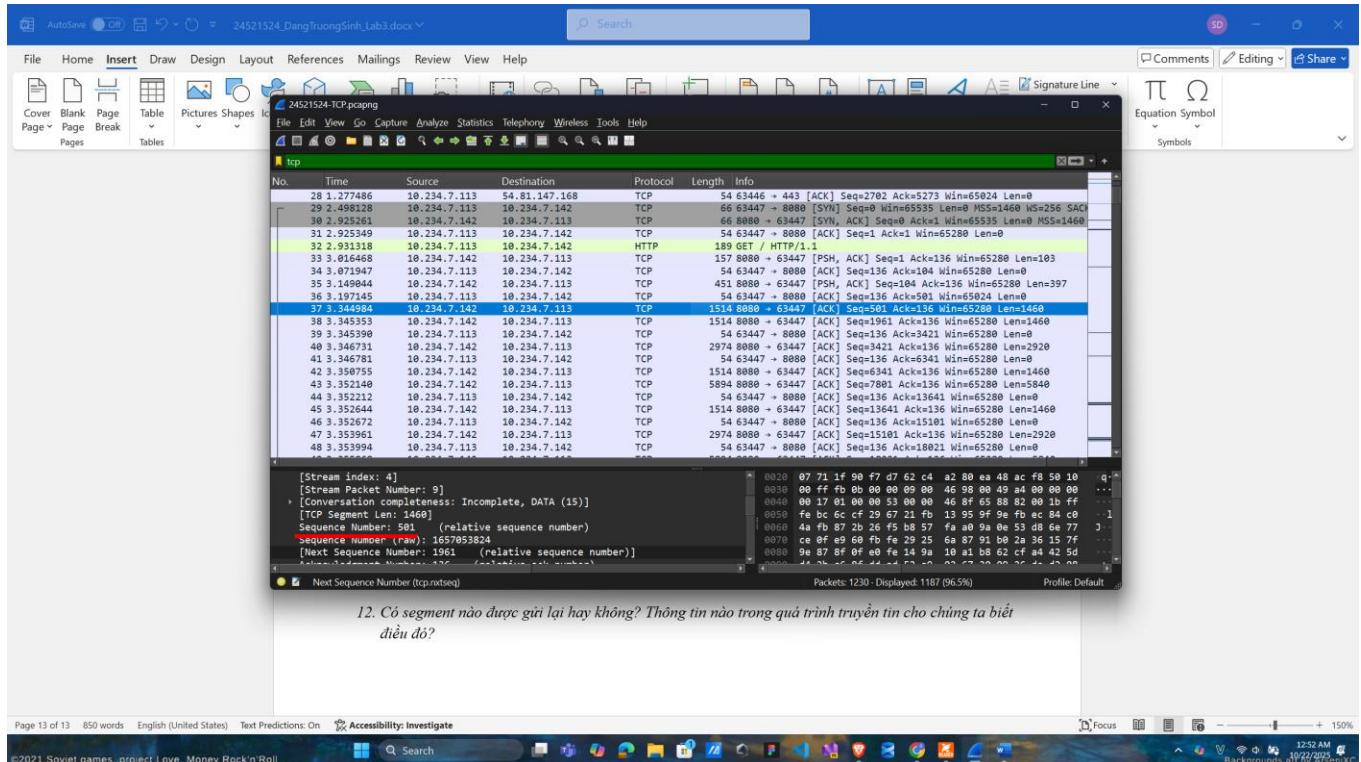
Hình 25. Sequence Number của Segment 30.



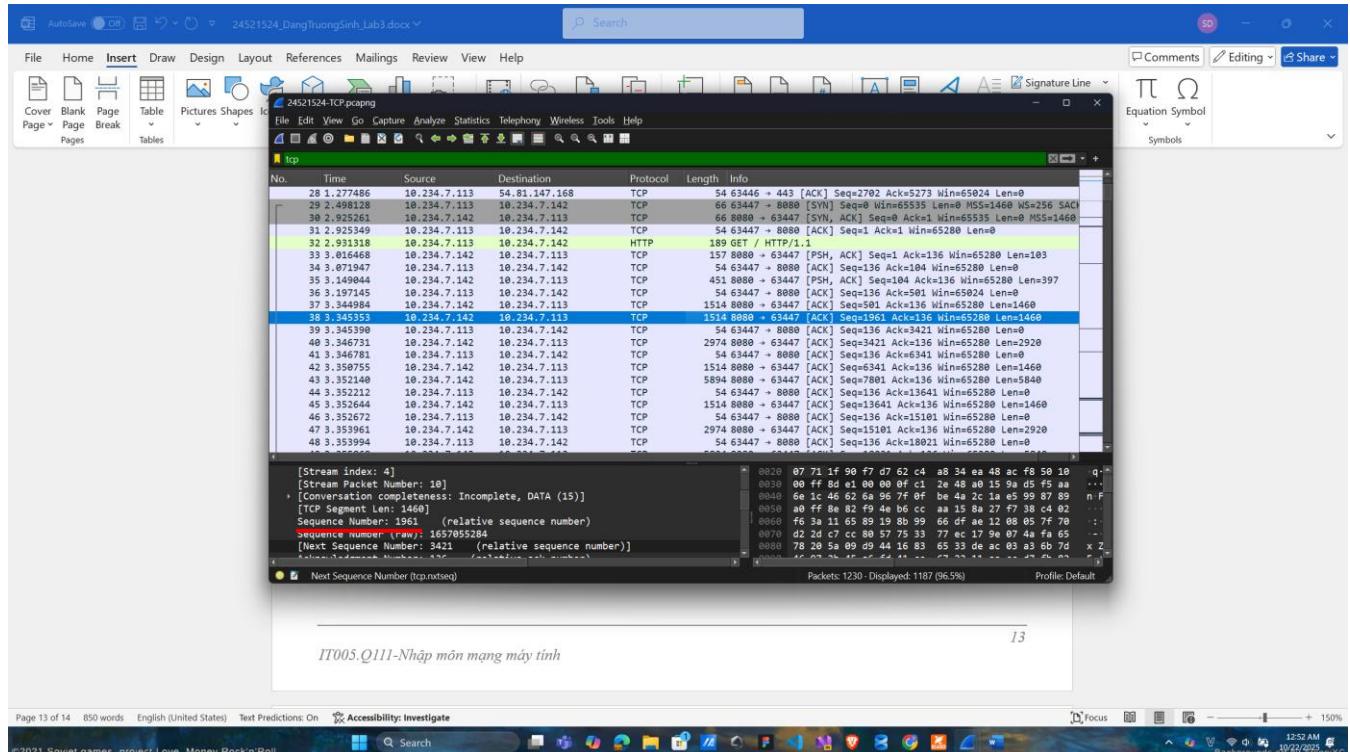
Hình 26. Sequence Number của Segment 33



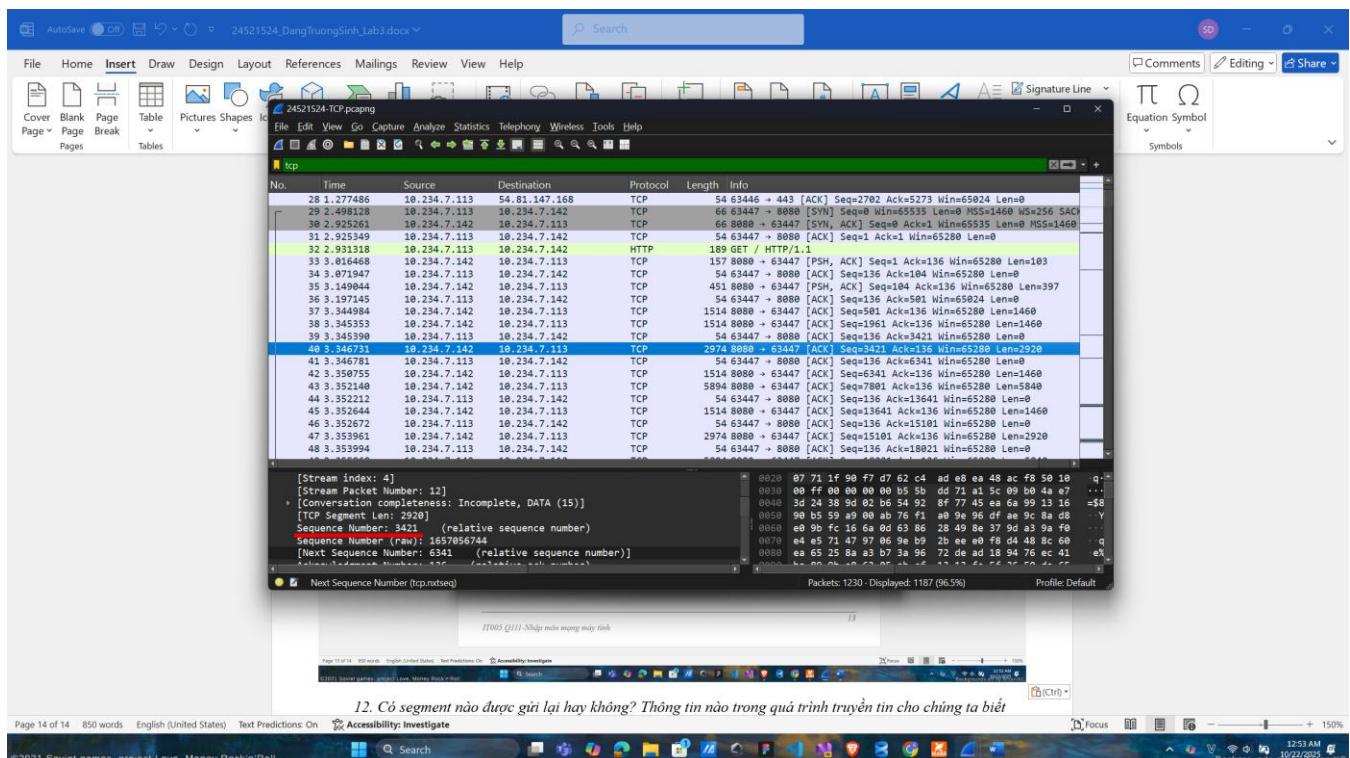
Hình 27. Sequence Number của Segment 35



Hình 28. Sequence Number của Segment 37

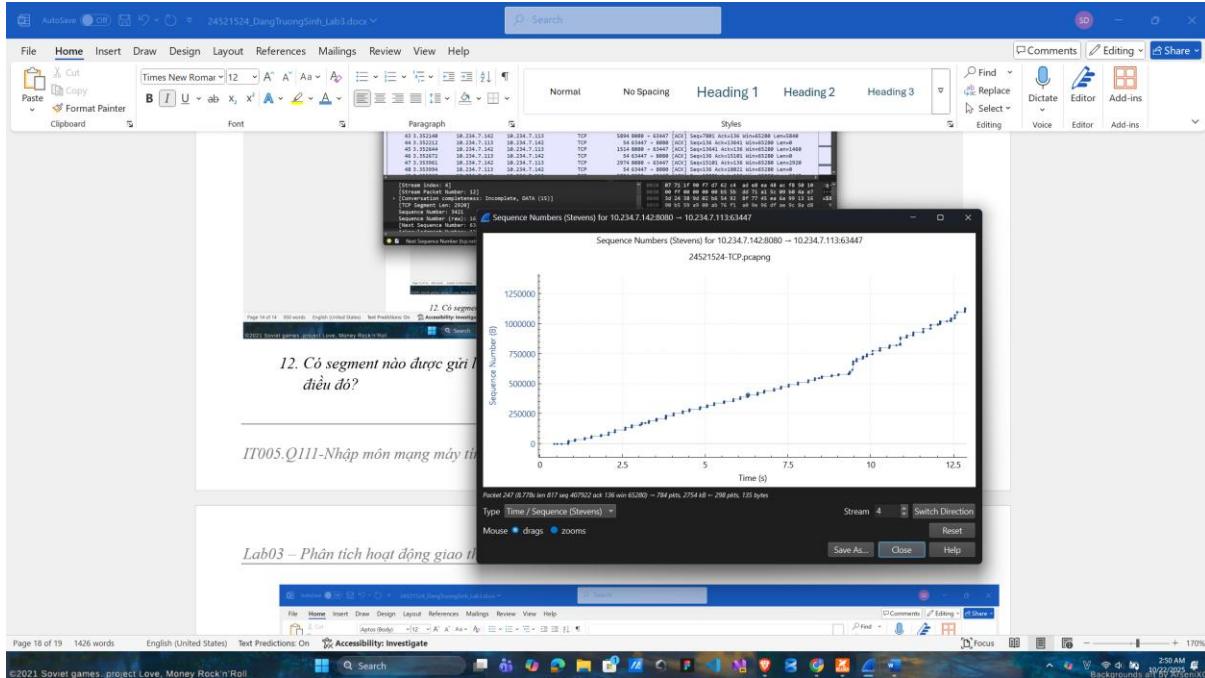


Hình 29. Sequence Number của Segment 38

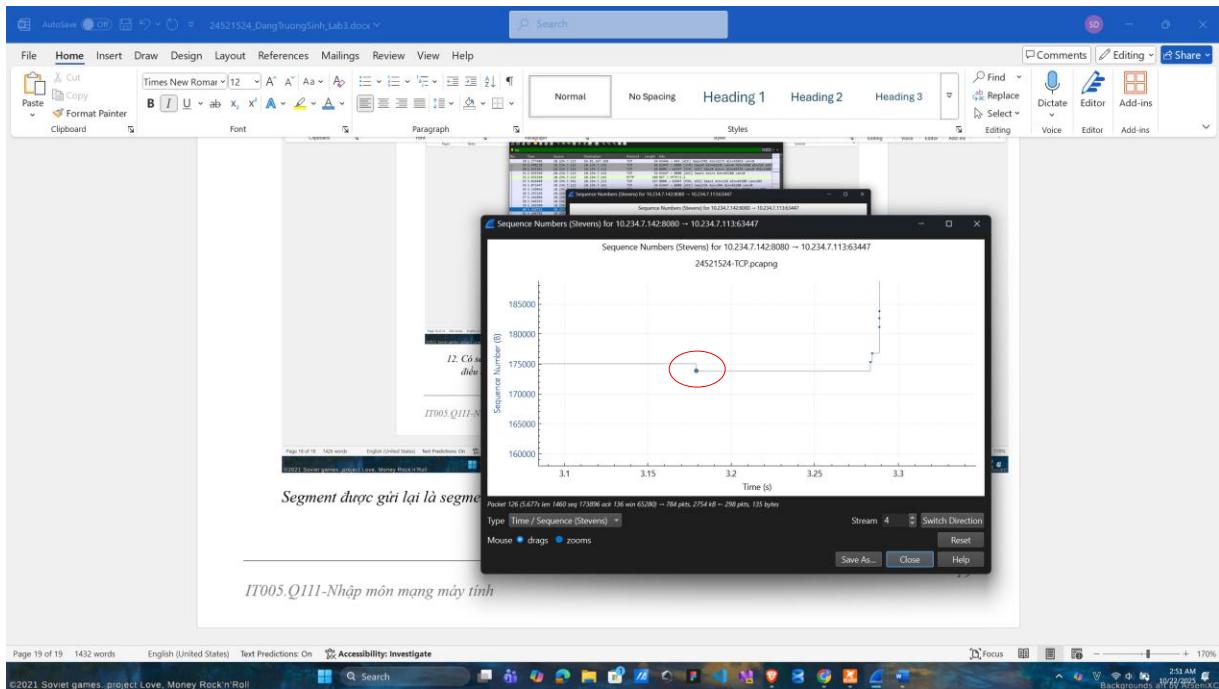


Hình 30. Sequence Number của Segment 40

12. Có segment nào được gửi lại hay không? Thông tin nào trong quá trình truyền tin cho chúng ta biết điều đó?



Hình 31. Biểu đồ Sequence Number (Stevens)



Hình 32. Hình ảnh segment bị gửi lại

Segment được gửi lại là segment 126

Trên đồ thị **Time–Sequence Graph (Stevens)**, nếu một segment bị gửi lại, đường biểu diễn sẽ giảm hoặc lặp lại tại cùng giá trị Sequence Number.