

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



MẠNG MÁY TÍNH TN (CO3094)

BÀI TẬP LAB 3A

WIRESHARK TCP

LỚP: L09

GVHD: Bùi Xuân Giang

Sinh viên thực hiện

Nguyễn Tấn Tài : 2212990

Thành phố Hồ Chí Minh, tháng 9 năm 2024

Trong bài lab này, chúng ta sẽ điều tra hành vi của giao thức TCP nổi tiếng một cách chi tiết. Chúng ta sẽ thực hiện việc này bằng cách phân tích các gói tin TCP được gửi và nhận trong quá trình truyền tệp 150KB (chứa văn bản của *Alice's Adventures in Wonderland* của Lewis Carroll) từ máy tính của bạn đến một máy chủ từ xa. Chúng ta sẽ nghiên cứu việc sử dụng số thứ tự (sequence) và số xác nhận (acknowledgment) của TCP để cung cấp quá trình truyền dữ liệu đáng tin cậy. Chúng ta sẽ xem xét thuật toán điều khiển nghẽn (congestion control) của TCP — khởi đầu chậm (slow start) và tránh nghẽn (congestion avoidance) — và sẽ xem xét cơ chế điều khiển luồng (flow control) được quảng cáo bởi người nhận (receiver-advertised). Chúng ta cũng sẽ xem xét cài đặt kết nối TCP và sẽ điều tra hiệu suất (thông lượng và thời gian khứ hồi) của kết nối TCP giữa máy tính của bạn và máy chủ.

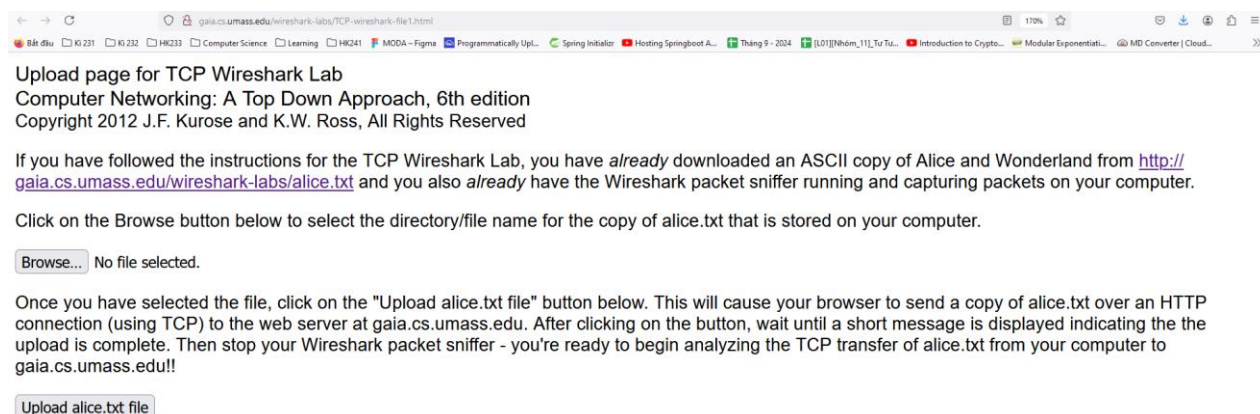
Trước khi bắt đầu bài lab này, bạn nên xem lại các mục 3.5 và 3.7 trong sách giáo khoa.

1. Ghi lại quá trình truyền tải TCP lớn từ máy tính của bạn đến máy chủ từ xa

Trước khi bắt đầu khám phá về TCP, chúng ta sẽ cần sử dụng Wireshark để thu được một bản ghi của quá trình truyền TCP của một tệp từ máy tính của bạn đến máy chủ từ xa. Bạn sẽ thực hiện việc này bằng cách truy cập một trang web cho phép bạn nhập tên của một tệp được lưu trữ trên máy tính của bạn (chứa văn bản ASCII của Alice in Wonderland) và sau đó truyền tệp này đến một máy chủ web bằng cách sử dụng phương thức HTTP POST (xem phần 2.2.3 trong sách). Chúng ta sẽ sử dụng phương thức POST thay vì phương thức GET vì chúng ta muốn truyền một lượng dữ liệu lớn từ máy tính của bạn đến máy tính khác. Tất nhiên, chúng ta sẽ chạy Wireshark trong thời gian này để thu được bản ghi của các đoạn TCP được gửi và nhận từ máy tính của bạn.

Các bước thực hiện

1. Mở trình duyệt web của bạn. Truy cập đường dẫn <http://gaia.cs.umass.edu/wireshark-labs/alice.txt> và tải về bản sao ASCII của Alice in Wonderland. Lưu tệp này ở đâu đó trên máy tính của bạn.
2. Tiếp theo truy cập <http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html>. Bạn sẽ thấy một màn hình giống như:



3. Nhấn nút *Browse* để chọn tệp Alice in Wonderland trên máy tính của bạn. **Đừng nhấn nút *Upload alice.txt file* ngay.**
4. Khởi động Wireshark và bắt đầu quá trình ghi lại các gói tin (Capture -> Start), sau đó nhấn OK trong màn hình tùy chọn Wireshark Packet Capture (chúng ta sẽ không cần chọn các tùy chọn khác).
5. Trở lại trình duyệt của bạn, nhấn nút *Upload alice.txt file* để tải tệp lên máy chủ gaia.cs.umass.edu. Sau khi tệp đã được tải lên, một thông báo chúc mừng ngắn sẽ được hiển thị trong cửa sổ trình duyệt của bạn.

Congratulations!

You've now transferred a copy of *alice.txt* from your computer to gaia.cs.umass.edu. You should now stop Wireshark packet capture. It's time to start analyzing the captured Wireshark packets!

6. Dừng việc ghi lại gói tin của Wireshark. Cửa sổ Wireshark của bạn sẽ giống với hình dưới đây.

Nếu bạn không thể chạy Wireshark trên kết nối mạng trực tiếp, bạn có thể tải về bản ghi các gói tin đã được ghi lại theo các bước trên trên một trong các máy tính của tác giả. Bạn sẽ thấy có giá trị khi tải về bản ghi này ngay cả khi bạn đã tự ghi lại bản ghi của mình và sử dụng nó, cùng với bản ghi của bạn, khi khám phá các câu hỏi bên dưới.

2. Xem trước bản ghi đã thu được

Trước khi phân tích hành vi của kết nối TCP một cách chi tiết, hãy xem qua bản ghi.

Đầu tiên, lọc các gói tin được hiển thị trong cửa sổ Wireshark bằng cách nhập "tcp" (chữ thường, không có dấu ngoặc kép và đừng quên nhấn phím Enter sau khi nhập!) vào cửa sổ lọc hiển thị ở gần đầu cửa sổ Wireshark.

Những gì bạn sẽ thấy là loạt thông điệp TCP và HTTP giữa máy tính của bạn và gaia.cs.umass.edu. Bạn sẽ thấy bắt đầu của bắt tay ba chiều (three-way handshake) bao gồm một thông điệp SYN. Bạn sẽ thấy một thông điệp HTTP POST. Tùy thuộc vào phiên bản Wireshark bạn đang sử dụng, bạn có thể thấy một loạt thông điệp "HTTP Continuation" được gửi từ máy tính của bạn đến gaia.cs.umass.edu.

Nhớ lại từ phần thảo luận trước về HTTP trong Wireshark, rằng không có khái niệm về một thông điệp HTTP Continuation — đây là cách Wireshark chỉ ra rằng có nhiều đoạn TCP được sử dụng để mang một thông điệp HTTP duy nhất. Trong các phiên bản mới hơn của Wireshark, bạn sẽ thấy "[TCP segment of a reassembled PDU]" trong cột Thông tin (Info) của hiển thị Wireshark để chỉ ra rằng đoạn TCP chứa dữ liệu thuộc về một thông điệp của tầng trên (trong trường hợp này là HTTP). Bạn cũng sẽ thấy các đoạn TCP ACK được gửi lại từ gaia.cs.umass.edu đến máy tính của bạn.

Hãy trả lời các câu hỏi sau, bằng cách mở tập tin gói tin đã được ghi lại trong Wireshark: tập tcp-ethereal-trace-1 tại <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip>. Tải tập này về và mở bản ghi trong Wireshark. Khi trả lời một câu hỏi, bạn nên nộp kèm theo bản in của gói tin (hoặc các gói tin) mà bạn đã sử dụng để trả lời câu hỏi đó. Chú thích trên bản in để giải thích câu trả lời của bạn. Để in một gói tin, sử dụng File -> Print, chọn Selected packet only, chọn Packet summary line và chọn lượng thông tin tối thiểu của chi tiết gói tin mà bạn cần để trả lời câu hỏi.

Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
1	20:44:20.570381	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
2	20:44:20.593553	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
3	20:44:20.593646	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	20:44:20.596858	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP PDU reassembled in 199]
5	20:44:20.612118	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP PDU reassembled in 199]

195	20:44:25.770633	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=160849 Ack=1 Win=17520 Len=1460 [TCP PDU reassembled in 199]
196	20:44:25.771531	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=162309 Ack=1 Win=17520 Len=1460 [TCP PDU reassembled in 199]
197	20:44:25.772405	192.168.1.102	128.119.245.12	TCP	326	1161 → 80 [PSH, ACK] Seq=163769 Ack=1 Win=17520 Len=272 [TCP PDU reassembled in 199]
198	20:44:25.867638	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=159389 Ack=62780 Len=0
199	20:44:25.867722	192.168.1.102	128.119.245.12	HTTP	104	POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1 (text/plain)

Trả lời câu hỏi

- Địa chỉ IP và cổng TCP của máy khách (máy tính nguồn) đang truyền tệp đến **gaia.cs.umass.edu** là gì? Để trả lời câu hỏi này, bạn có thể dễ dàng chọn một thông điệp HTTP và xem xét chi tiết của gói tin TCP được sử dụng để mang thông điệp HTTP này, bằng cách sử dụng "details of the selected packet header window" (tham chiếu đến Hình 2 trong phần "Getting Started with Wireshark" nếu bạn chưa quen thuộc với các cửa sổ của Wireshark).

```
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Destination: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Source: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Type: IPv4 (0x0800)
[Stream index: 0]
```

```
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 164041, Ack: 1, Len: 50
```

Như ta quan sát được:

- Source IP Address:** Là địa chỉ IP của máy khách, **Source IP Address: 192.168.1.102**
- Source Port:** Là cổng TCP của máy khách, **Source Port: 1161**

- Địa chỉ IP của **gaia.cs.umass.edu** là gì? Nó đang sử dụng cổng nào để gửi và nhận các đoạn TCP cho kết nối này?

No.	Time	Source	Destination	Protocol	Length	Info
199	20:44:25.867722	192.168.1.102	128.119.245.12	HTTP	104	POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1 (text/plain)

- IP của **gaia.cs.umass.edu**: 128.119.245.12**

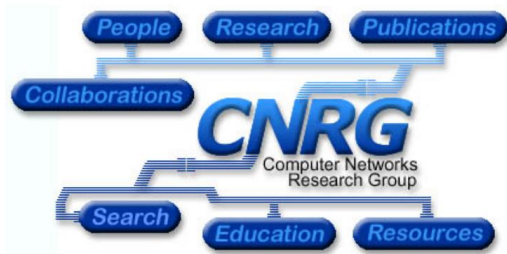
```
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Destination: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Source: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Type: IPv4 (0x0800)
[Stream index: 0]
```

```
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
```

- Cổng TCP: Cổng 80.**

Nếu bạn có thể tự tạo ra bản ghi của mình, hãy trả lời câu hỏi sau:

3. Địa chỉ IP và cổng TCP của máy khách (máy tính nguồn) được sử dụng để truyền tệp đến gaia.cs.umass.edu là gì?



No.	Time	Source	Destination	Protocol	Length	Info
6344	14:49:48.505859	192.168.88.159	128.119.245.12	TCP	66	63162 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
6345	14:49:48.506112	192.168.88.159	128.119.245.12	TCP	66	63163 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
6350	14:49:48.757873	192.168.88.159	128.119.245.12	TCP	66	63164 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
6360	14:49:48.782465	192.168.88.159	128.119.245.12	TCP	66	63165 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
6362	14:49:48.809184	128.119.245.12	192.168.88.159	TCP	66	443 → 63163 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=128

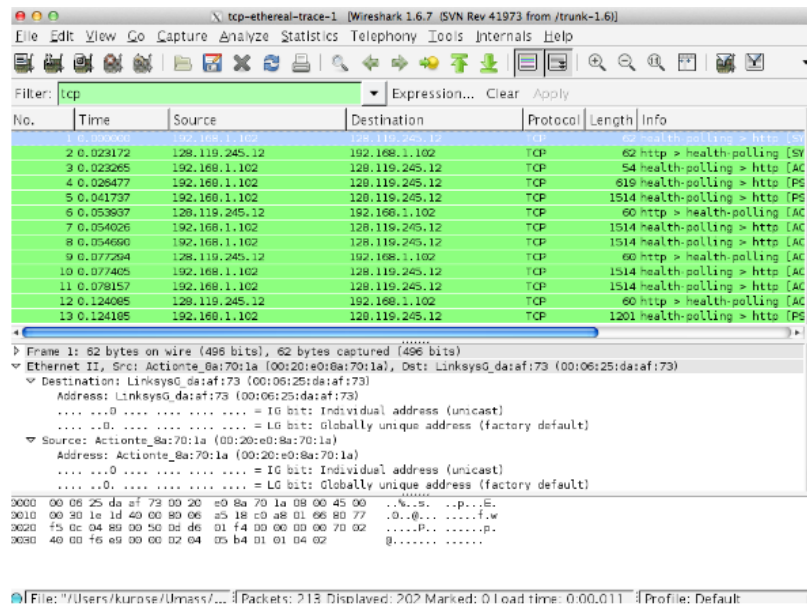
```

No.      Time      Source      Destination  Protocol Length Info
 6344 14:49:48.505859 192.168.88.159 128.119.245.12 TCP        66      63162 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460
WS=256 SACK_PERM
Frame 6344: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{E8972303-5AC7-4ADA-9205-681936D705B5},
id 0
Ethernet II, Src: Intel_db:97:af (00:93:37:db:97:af), Dst: VietnamPostA_f2:d0:ce (cc:71:90:f2:d0:ce)
Internet Protocol Version 4, Src: 192.168.88.159, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 63162, Dst Port: 80, Seq: 0, Len: 0
[Community ID: 1:d6nIFGHf5bNokOdB9aOfZ4qZ6w=]
TRANSMISSION RTE Data
  
```

Địa chỉ IP của máy khách: 192.168.88.159, đây sẽ là địa chỉ IP của máy tính của bạn.

Cổng TCP của máy khách: 63162, đây là cổng TCP được chọn ngẫu nhiên bởi máy tính để khởi tạo kết nối với máy chủ.

Vì bài lab này là về TCP chứ không phải HTTP, hãy thay đổi cửa sổ "danh sách các gói tin đã thu được" của Wireshark sao cho nó hiển thị thông tin về các đoạn TCP chứa các thông điệp HTTP, thay vì hiển thị về các thông điệp HTTP. Để Wireshark làm điều này, hãy chọn Analyze -> Enabled Protocols. Sau đó bỏ chọn hộp HTTP và nhấn OK. Bạn bây giờ sẽ thấy cửa sổ Wireshark như sau:



The image shows a Wireshark capture of TCP traffic. The packet list shows several TCP segments. The packet details pane for the selected packet (No. 1) shows the Ethernet II header and the IP header. The IP header shows the source IP as 192.168.1.102 and the destination IP as 128.119.245.12. The packet bytes pane shows the raw data of the packet.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000	192.168.1.102	128.119.245.12	TCP	62	health-polling > http [SYN]
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	http > health-polling [SYN]
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	health-polling > http [ACK]
4	0.028477	192.168.1.102	128.119.245.12	TCP	619	health-polling > http [PSH]
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [ACK]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	http > health-polling [ACK]
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [ACK]
8	0.054660	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [ACK]
9	0.077264	128.119.245.12	192.168.1.102	TCP	60	http > health-polling [ACK]
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [ACK]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [ACK]
12	0.124065	128.119.245.12	192.168.1.102	TCP	60	http > health-polling [ACK]
13	0.124105	192.168.1.102	128.119.245.12	TCP	1201	health-polling > http [PSH]

Đây là những gì chúng ta đang tìm kiếm - một loạt các phân đoạn TCP được gửi giữa máy tính của bạn và `gaia.cs.umass.edu`. Chúng ta sẽ sử dụng dấu vết gói tin mà bạn đã chụp (và/hoặc dấu vết gói tin `tcp-ethereal-trace-1` trong `http://gaia.cs.umass.edu/wireshark-labs/wireshark-trace.zip`; xem chú thích trước đó) để nghiên cứu hành vi TCP trong phần còn lại của phòng thí nghiệm này.

3. Cơ bản về TCP

Trả lời các câu hỏi sau đây cho các đoạn TCP:

4. Số thứ tự (sequence number) của đoạn TCP SYN được sử dụng để khởi tạo kết nối TCP giữa máy tính khách và `gaia.cs.umass.edu` là gì? Trong đoạn này, yếu tố nào xác định rằng đây là một đoạn SYN?

Chọn gói tin có cột Info chứa dòng "[SYN]" (gói tin đầu tiên trong bắt tay ba bước của TCP). Trong phần Packet Details, mở rộng mục Transmission Control Protocol (TCP) để xem Sequence Number. Xác định đoạn SYN: Một đoạn SYN sẽ có cờ SYN được bật trong trường Flags. Tìm cờ SYN trong phần Flags của TCP, thường là giá trị `SYN = 1`.

```
tcp.flags.syn == 1 && tcp.flags.ack == 0
```

No.	Time	Source	Destination	Protocol	Length	Info
1	20:44:20.579381	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)						
Ethernet II, Src: ActiontecEle_8a:70:1a (08:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)						
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12						
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0						
[Community ID: 1:genXGslpfaEf/y4DMtn0xLnIyE=]						
TRANSMISSION Data						

Số thứ tự (sequence number) của đoạn TCP SYN là 0.

Để xác định rằng đây là một đoạn SYN, kiểm tra phần Flags trong trường TCP

- SYN = 1: Cờ SYN sẽ được bật (giá trị là 1), điều này chỉ ra rằng đây là một gói SYN.
- ACK = 0: Cờ ACK sẽ không được bật, do đây là gói tin đầu tiên trong quá trình bắt tay ba bước, chưa có ACK từ máy khách.

5. Số thứ tự của đoạn SYNACK được gửi bởi gaia.cs.umass.edu đến máy tính khách để phản hồi SYN là gì? Giá trị của trường xác nhận (Acknowledgement field) trong đoạn SYNACK là gì? Làm thế nào mà gaia.cs.umass.edu xác định giá trị đó? Yếu tố nào trong đoạn này xác định rằng đây là một đoạn SYNACK?

Sử dụng bộ lọc sau để hiển thị các gói SYN-ACK

```
tcp.flags.syn == 1 && tcp.flags.ack == 1
```

```
No.      Time                Source              Destination          Protocol Length Info
  2 20:44:20.593553    128.119.245.12      192.168.1.102        TCP                62      80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0
MSS=1460 SACK_PERM
Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
Ethernet II, Src: LinksysGroup_da:af:73 (00:06:25:da:af:73), Dst: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
[Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]
```

Số thứ tự (Sequence Number) của gói tin SYN-ACK: 0.

Giá trị của Acknowledgment Number là 1, bằng số thứ tự của gói tin SYN từ máy khách cộng thêm 1. Điều này có nghĩa là máy chủ gaia.cs.umass.edu xác nhận đã nhận được gói SYN của máy khách.

Để xác định đây là đoạn SYN-ACK, kiểm tra các cờ trong phần Flags của TCP

- SYN = 1: Cờ SYN được bật, cho biết đây là một phần của quá trình bắt tay.
- ACK = 1: Cờ ACK cũng được bật, cho thấy máy chủ đang xác nhận việc nhận được gói tin SYN của máy khách.

6. Số thứ tự của đoạn TCP chứa lệnh HTTP POST là gì? Lưu ý rằng để tìm lệnh POST, bạn cần đào sâu vào nội dung gói tin ở phần cuối của sổ Wireshark, tìm một đoạn có chứa "POST" trong trường Dữ liệu (DATA field) của nó.

```
http.request.method == "POST"
```



```

No.      Time                Source          Destination    Protocol Length Info
  199  20:44:25.867722      192.168.1.102    128.119.245.12  HTTP      104    POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
(text/plain)
Frame 199: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 164041, Ack: 1, Len: 50
[122 Reassembled TCP Segments (164090 bytes): #4(565), #5(1460), #7(1460), #8(1460), #10(1460), #11(1460), #13(1147), #18(1460), #19(1460), #20(1460), #21(1460), #22(1460), #23(892), #30(1460), #31(1460), #32(1460), #33(1460), #34(1460), ]
Hypertext Transfer Protocol
MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----265001916915724"
[Community ID: 1:genwXGslpfaEf/y4DMtn0xLnIyE=]
TRANSMUTE Data

```

Số thứ tự (Sequence Number) của gói tin: 164041.

TCP segment of a reassembled PDU, những gói này cho biết dữ liệu POST được truyền qua nhiều đoạn TCP.

7. Xem xét đoạn TCP chứa lệnh HTTP POST là đoạn đầu tiên trong kết nối TCP. Số thứ tự của sáu đoạn đầu tiên trong kết nối TCP (bao gồm cả đoạn chứa lệnh HTTP POST) là gì? Thời điểm nào từng đoạn được gửi? Khi nào ACK của từng đoạn được nhận? Xét sự chênh lệch giữa thời điểm mà mỗi đoạn TCP được gửi và khi nhận được xác nhận của nó, giá trị RTT (Round Trip Time) cho mỗi đoạn là gì? Giá trị RTT ước tính (EstimatedRTT) là gì (xem phần 3.5.3, trang 242 trong sách)? Giả sử giá trị của EstimatedRTT bằng với giá trị RTT đo được cho đoạn đầu tiên, và sau đó được tính bằng cách sử dụng phương trình EstimatedRTT trên trang 242 cho tất cả các đoạn sau đó.

Tương tự câu ở trên, Số thứ tự (Sequence Number) của gói tin: 164041.

```

No.      Time                Source          Destination    Protocol Length Info
  1  20:44:20.570381      192.168.1.102    128.119.245.12  TCP        62    1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460
SACK_PERM
Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
[Community ID: 1:genwXGslpfaEf/y4DMtn0xLnIyE=]
TRANSMUTE Data

No.      Time                Source          Destination    Protocol Length Info
  2  20:44:20.593553      128.119.245.12    192.168.1.102    TCP        62    80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0
MSS=1460 SACK_PERM
Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
Ethernet II, Src: LinksysGroup_da:af:73 (00:06:25:da:af:73), Dst: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
[Community ID: 1:genwXGslpfaEf/y4DMtn0xLnIyE=]

```



```
No.      Time      Source      Destination      Protocol Length Info
3 20:44:20.593646 192.168.1.102 128.119.245.12   TCP      54      1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
Frame 3: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
[Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]
```

```
No.      Time      Source      Destination      Protocol Length Info
4 20:44:20.596858 192.168.1.102 128.119.245.12   TCP      619     1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520
Len=565 [TCP PDU reassembled in 199]
Frame 4: 619 bytes on wire (4952 bits), 619 bytes captured (4952 bits)
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565
Source Port: 1161
Destination Port: 80
[Stream index: 0]
[Stream Packet Number: 4]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 565]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 232129013
[Next Sequence Number: 566 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 883061786
0101 ... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 17520
[Calculated window size: 17520]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x1fbd [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
TCP payload (565 bytes)
[Reassembled PDU in frame: 199]
TCP segment data (565 bytes)
[Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]
```

```
No.      Time      Source      Destination      Protocol Length Info
199 20:44:25.867722 192.168.1.102 128.119.245.12   HTTP      104     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
(text/plain)
Frame 199: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 164041, Ack: 1, Len: 50
Source Port: 1161
Destination Port: 80
[Stream index: 0]
[Stream Packet Number: 196]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 50]
Sequence Number: 164041 (relative sequence number)
Sequence Number (raw): 232293053
[Next Sequence Number: 164091 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 883061786
0101 ... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 17520
[Calculated window size: 17520]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x9f0f [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
TCP payload (50 bytes)
TCP segment data (50 bytes)
[122 Reassembled TCP Segments (164090 bytes): #4(565), #5(1460), #7(1460), #8(1460), #10(1460), #11(1460), #13(1147), #18(1460), #19(1460), #20(1460), #21(1460), #22(1460), #23(892), #30(1460), #31(1460), #32(1460), #33(1460), #34(1460), ]
Hypertext Transfer Protocol
MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----265001916915724"
[Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]
TRANSUM RTE Data
```

Lưu ý: Wireshark có tính năng tiện lợi cho phép bạn vẽ biểu đồ RTT cho từng đoạn TCP đã gửi. Chọn một đoạn TCP trong cửa sổ "danh sách các gói đã thu" đang được gửi từ máy khách đến máy chủ giaia.cs.umass.edu. Sau đó chọn Statistics -> TCP Stream Graph -> Round Trip Time Graph.

8. Chiều dài của mỗi đoạn trong sáu đoạn TCP đầu tiên là bao nhiêu?

No.	Time	Source	Destination	Protocol	Length	Info
1	20:44:20.570381	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73) Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12 Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0 [Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=] TRANSMISSION RTE Data						
2	20:44:20.593553	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) Ethernet II, Src: LinksysGroup_da:af:73 (00:06:25:da:af:73), Dst: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a) Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102 Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0 [Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]						
3	20:44:20.593646	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
Frame 3: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73) Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12 Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 0 [Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]						
4	20:44:20.596858	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP PDU reassembled in 199]
Frame 4: 619 bytes on wire (4952 bits), 619 bytes captured (4952 bits) Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73) Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12 Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565 Source Port: 1161 Destination Port: 80 [Stream index: 0] [Stream Packet Number: 4] [Conversation completeness: Incomplete, DATA (15)] [TCP Segment Len: 565] Sequence Number: 1 (relative sequence number) Sequence Number (raw): 232129013 [Next Sequence Number: 566 (relative sequence number)] Acknowledgment Number: 1 (relative ack number) Acknowledgment number (raw): 883061786 0101 ... = Header Length: 20 bytes (5) Flags: 0x018 (PSH, ACK) Window: 17520 [Calculated window size: 17520] [Window size scaling factor: -2 (no window scaling used)] Checksum: 0x1fbd [unverified] [Checksum Status: Unverified] Urgent Pointer: 0 [Timestamps] [SEQ/ACK analysis] TCP payload (565 bytes) [Reassembled PDU in frame: 199] TCP segment data (565 bytes) [Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]						

No.	Time	Source	Destination	Protocol	Length	Info
199	20:44:25.867722	192.168.1.102	128.119.245.12	HTTP	104	POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1

(text/plain)
Frame 199: 104 bytes on wire (832 bits), 104 bytes captured (832 bits) on interface
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 164041, Ack: 1, Len: 50
Source Port: 1161
Destination Port: 80
[Stream index: 0]
[Stream Packet Number: 196]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 50]
Sequence Number: 164041 (relative sequence number)
Sequence Number (raw): 232293053
[Next Sequence Number: 164091 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 883061786
0101 = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 17520
[Calculated window size: 17520]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x9f0f [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
TCP payload (50 bytes)
TCP segment data (50 bytes)
[122 Reassembled TCP Segments (164090 bytes): #4(565), #5(1460), #7(1460), #8(1460), #10(1460), #11(1460), #13(1147), #18(1460), #19(1460), #20(1460), #21(1460), #22(1460), #23(892), #30(1460), #31(1460), #32(1460), #33(1460), #34(1460),]
Hypertext Transfer Protocol
MIME multipart media encapsulation, Type: multipart/form-data, Boundary: "-----265001916915724"
[Community ID: 1:genwXGslpfaEf/y4DMtn0xLnIyE=]
TRANSMISSION RTE Data

9. Lượng bộ đệm có sẵn tối thiểu được quảng cáo tại phía nhận trong toàn bộ bản ghi là bao nhiêu? Việc thiếu không gian bộ đệm có bao giờ gây nghẽn đối với người gửi không?

Để xác định Window Size được quảng cáo từ phía nhận, ta có thể lọc các gói tin ACK mà máy chủ (IP 128.119.245.12) gửi cho máy khách (IP 192.168.1.102).

tcp.analysis.ack_rtt && ip.src == 128.119.245.12

No.	Time	Source	Destination	Protocol	Length	Info
6	20:44:20.624318	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=566 Win=6780 Len=0

Frame 6: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface
Ethernet II, Src: LinksysGroup_da:af:73 (00:06:25:da:af:73), Dst: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 1, Ack: 566, Len: 0
Source Port: 80
Destination Port: 1161
[Stream index: 0]
[Stream Packet Number: 6]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 0]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 883061786
[Next Sequence Number: 1 (relative sequence number)]
Acknowledgment Number: 566 (relative ack number)
Acknowledgment number (raw): 232129578
0101 = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
Window: 6780
[Calculated window size: 6780]
[Window size scaling factor: -2 (no window scaling used)]
Checksum: 0x9e30 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
[Community ID: 1:genwXGslpfaEf/y4DMtn0xLnIyE=]

Window Size bằng 0: Nếu tại bất kỳ thời điểm nào, Window Size bằng 0, điều đó có nghĩa là bộ đệm phía nhận đã đầy và không thể nhận thêm dữ liệu từ phía gửi. Khi điều này xảy ra, máy gửi sẽ phải dừng việc truyền dữ liệu cho đến khi phía nhận thông báo rằng bộ đệm đã trống trở lại (tức là Window Size sẽ được tăng lên).

```
tcp.window_size == 0
```

No.	Time	Source	Destination	Protocol	Length	Info
-----	------	--------	-------------	----------	--------	------

Điều này cho thấy bộ đệm chưa đầy và việc truyền dữ liệu từ máy gửi vẫn tiếp tục.

10. Có đoạn nào bị truyền lại trong bản ghi không? Bạn đã kiểm tra những gì (trong bản ghi) để trả lời câu hỏi này?

Sử dụng bộ lọc để tìm các đoạn bị truyền lại. Nhập bộ lọc sau vào thanh lọc của Wireshark

```
tcp.analysis.retransmission || tcp.analysis.fast_retransmission  
|| tcp.analysis.duplicate_ack
```

Bộ lọc này sẽ hiển thị tất cả các gói tin TCP bị truyền lại. Các đoạn này là những gói tin mà Wireshark phát hiện bị truyền lại do không nhận được ACK từ phía nhận hoặc do xảy ra lỗi khác.

No.	Time	Source	Destination	Protocol	Length	Info
-----	------	--------	-------------	----------	--------	------

Từ hình ảnh ở trên, không có đoạn nào bị truyền lại trong bản ghi.

11. Thông thường bộ nhận xác nhận bao nhiêu dữ liệu trong một ACK? Bạn có thể xác định trường hợp mà bộ nhận xác nhận sau mỗi đoạn nhận được không (xem Bảng 3.2 trên trang 250 trong sách)?

No.	Time	Source	Destination	Protocol	Length	Info
2	20:44:20.593553	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
6	20:44:20.624318	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0


```

No.      Time      Source      Destination  Protocol Length Info
 6 20:44:20.624318 128.119.245.12 192.168.1.102 TCP        60      80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
Frame 6: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
Ethernet II, Src: LinksysGroup_da:af:73 (00:06:25:da:af:73), Dst: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 1, Ack: 566, Len: 0
  Source Port: 80
  Destination Port: 1161
  [Stream index: 0]
  [Stream Packet Number: 6]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 0]
  Sequence Number: 1 (relative sequence number)
  Sequence Number (raw): 883061786
  [Next Sequence Number: 1 (relative sequence number)]
  Acknowledgment Number: 566 (relative ack number)
  Acknowledgment number (raw): 232129578
  0101 ... = Header Length: 20 bytes (5)
  Flags: 0x010 (ACK)
  Window: 6780
  [Calculated window size: 6780]
  [Window size scaling factor: -2 (no window scaling used)]
  Checksum: 0x9e30 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
  [Timestamps]
  [SEQ/ACK analysis]
[Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]

```

Acknowledgment Number cho biết byte tiếp theo mà máy chủ mong đợi từ máy khách, vì vậy lượng dữ liệu mà máy chủ đã nhận được là từ byte 1 đến byte 565. Do đó, lượng dữ liệu được xác nhận trong gói tin ACK này là **565 bytes**.

```

No.      Time      Source      Destination  Protocol Length Info
 5 20:44:20.612118 192.168.1.102 128.119.245.12 TCP      1514    1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520
Len=1460 [TCP PDU reassembled in 199]
Frame 5: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)
Ethernet II, Src: ActiontecEle_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysGroup_da:af:73 (00:06:25:da:af:73)
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 566, Ack: 1, Len: 1460
  Source Port: 1161
  Destination Port: 80
  [Stream index: 0]
  [Stream Packet Number: 5]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 1460]
  Sequence Number: 566 (relative sequence number)
  Sequence Number (raw): 232129578
  [Next Sequence Number: 2026 (relative sequence number)]
  Acknowledgment Number: 1 (relative ack number)
  Acknowledgment number (raw): 883061786
  0101 ... = Header Length: 20 bytes (5)
  Flags: 0x018 (PSH, ACK)
  Window: 17520
  [Calculated window size: 17520]
  [Window size scaling factor: -2 (no window scaling used)]
  Checksum: 0x3be5 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
  [Timestamps]
  [SEQ/ACK analysis]
  TCP payload (1460 bytes)
  [Reassembled PDU in frame: 199]
  TCP segment data (1460 bytes)
[Community ID: 1:genWXGslpfaEf/y4DMtn0xLnIyE=]

```

Đây là gói trước nó, Gói tin này có Sequence Number 566 và gửi 1460 bytes dữ liệu.

Máy chủ đã nhận tất cả dữ liệu trước đó (đến byte 565), và gói ACK tiếp theo từ máy chủ sẽ xác nhận rằng nó đã nhận dữ liệu từ byte 566 đến 2025.

12. Thông lượng (số byte được truyền trên mỗi đơn vị thời gian) của kết nối TCP là bao nhiêu? Giải thích cách bạn tính giá trị này.

$$\text{Thông lượng} = \frac{\text{Tổng số byte truyền}}{\text{Tổng thời gian truyền}}$$

Sử dụng bộ lọc để chỉ hiển thị các gói tin dữ liệu từ máy khách đến máy chủ (giả sử máy khách là 192.168.1.102 và máy chủ là 128.119.245.12)

Ở thanh menu, chọn Statistics -> Conversations. Trong cửa sổ hiện ra, chọn tab TCP. Ở đây, ta sẽ thấy thông tin về tất cả các cuộc trò chuyện TCP (TCP connections) đang diễn ra trong bản ghi. Trong cột **Bytes**, bạn sẽ thấy tổng số byte đã được truyền trong từng kết nối. Bạn có thể xác định tổng số byte đã được truyền từ máy khách đến máy chủ hoặc ngược lại.

Ethernet · 3		IPv4 · 3		IPv6		TCP · 2		UDP · 1							
Address A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A	Flows
192.168.1.102	1161	128.119.245.12	80	201	176 kB	0	125	171 kB	76	5 kB	0.000000	5.6511	241 kbps	7483 bits/s	2
192.168.1.102	1162	199.2.53.206	631	1	62 bytes	1	1	62 bytes	0	0 bytes	7.595557	0.0000			0

Tổng số byte truyền (Bytes A → B): 171 KB (171 * 1024 = 174,976 bytes).

Thời gian truyền (Duration): 5.6511 giây.

Thông lượng (Bits/s A → B): 241 kbps (kết quả hiển thị sẵn từ Wireshark).

Thông lượng bằng byte/giây

$$\text{Throughput} = \frac{174,976 \text{ bytes}}{5.6511 \text{ giây}} = 30,964 \frac{\text{bytes}}{\text{giây}}$$

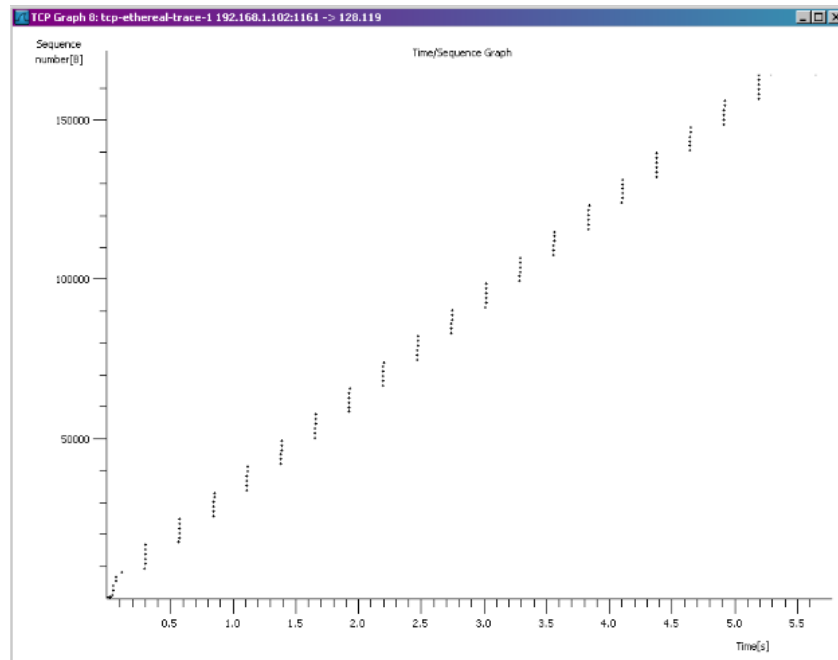
Thông lượng bằng bit/giây

$$\text{Throughput} \left(\frac{\text{bits}}{\text{s}} \right) = 30,964 \frac{\text{bytes}}{\text{giây}} \times 8 = 247,712 \frac{\text{bits}}{\text{giây}} \approx 241 \text{ kbps}$$

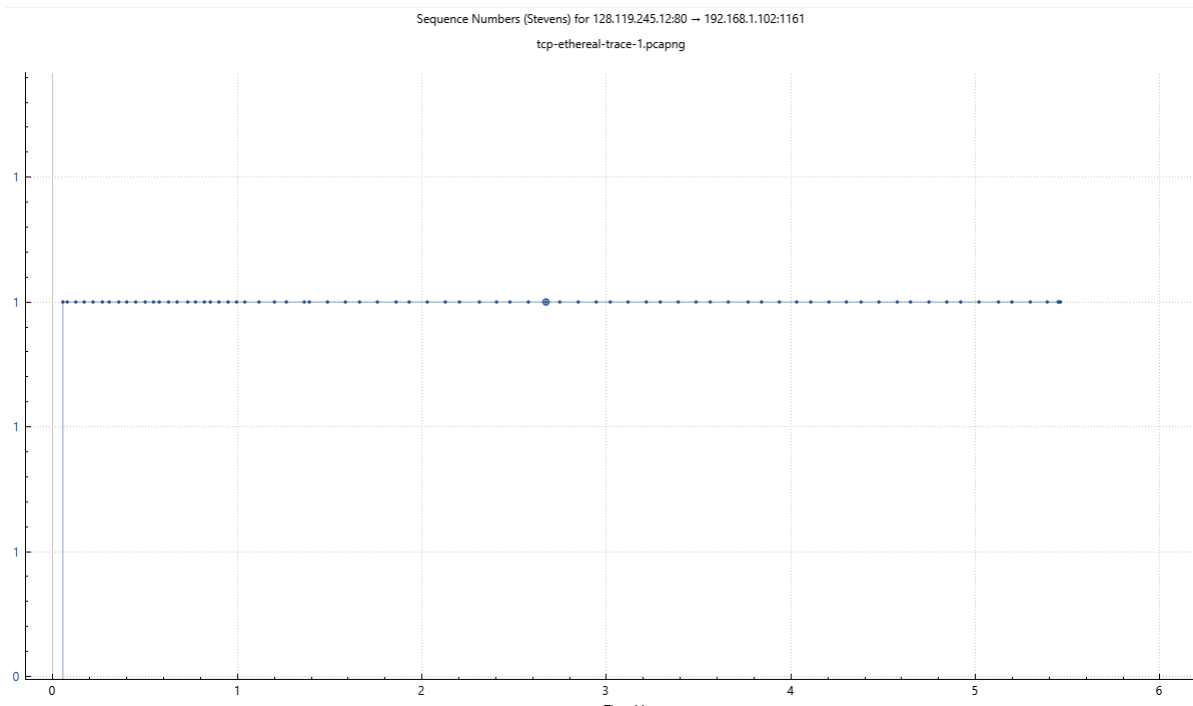
4. Điều khiển tắc nghẽn TCP trong thực tế

Bây giờ, chúng ta sẽ xem xét lượng dữ liệu được gửi trên mỗi đơn vị thời gian từ máy khách đến máy chủ. Thay vì (một cách tẻ nhạt) tính toán điều này từ dữ liệu thô trong cửa sổ Wireshark, chúng ta sẽ sử dụng một trong các tiện ích đồ thị của Wireshark - Time-Sequence-Graph (Stevens) - để vẽ dữ liệu.

Chọn một đoạn TCP trong cửa sổ "danh sách các gói đã thu" của Wireshark. Sau đó, chọn menu: *Statistics -> TCP Stream Graph -> Time-Sequence-Graph (Stevens)*. Bạn sẽ thấy một đồ thị tương tự như đồ thị dưới đây, được tạo từ các gói tin đã thu được trong bản ghi *tcp-ethereal-trace-1* tại <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip> (xem ghi chú phía trước).



Mỗi dấu chấm đại diện cho một đoạn TCP được gửi, vẽ số thứ tự của đoạn so với thời điểm nó được gửi. Lưu ý rằng một nhóm các dấu chấm được xếp chồng lên nhau đại diện cho một loạt các gói tin được gửi liên tiếp bởi người gửi.



13. Sử dụng công cụ Time-Sequence-Graph (Stevens) để xem đồ thị số thứ tự so với thời gian của các đoạn được gửi từ máy khách đến máy chủ gaia.cs.umass.edu. Bạn có thể xác định giai đoạn khởi đầu chậm (slowstart) của TCP bắt đầu và kết thúc ở đâu không, và khi nào điều khiển tránh tắc nghẽn (congestion avoidance) bắt đầu? Bình luận về những điểm khác biệt giữa dữ liệu đo lường được và hành vi lý tưởng của TCP mà chúng ta đã học trong sách.

Khởi đầu chậm (Slow Start) Giai đoạn khởi đầu chậm là khi TCP bắt đầu gửi một lượng nhỏ dữ liệu (thường là một gói tin), sau đó tăng dần theo cấp số nhân khi nhận được ACK từ phía nhận. Trong đồ thị lý tưởng, giai đoạn này sẽ được thể hiện dưới dạng một đoạn đường dốc tăng dần nhanh chóng.

Tránh tắc nghẽn (Congestion Avoidance): Sau khi đạt đến ngưỡng nhất định (thường là khi nhận thấy có sự mất gói hoặc bộ đệm đã gần đầy), TCP sẽ chuyển sang giai đoạn tránh tắc nghẽn, trong đó tốc độ truyền sẽ tăng chậm hơn (tăng theo tỷ lệ tuyến tính). Trong đồ thị, giai đoạn này sẽ biểu diễn dưới dạng một đoạn dốc ít hơn so với slow start.

Dựa vào đồ thị

- Giai đoạn khởi đầu chậm: Trong đoạn đầu tiên (từ khoảng thời gian 0 đến 1 giây), số thứ tự tăng rất ít và đều đặn. Điều này có thể cho thấy giai đoạn slow start, khi TCP tăng dần lượng dữ liệu gửi đi một cách từ từ và nhanh dần khi nhận được các ACK từ máy chủ.
- Giai đoạn tránh tắc nghẽn: Từ khoảng thời gian 2 giây trở đi, số thứ tự vẫn giữ nguyên (có vẻ không tăng), điều này có thể cho thấy mạng bắt đầu gặp vấn đề về tắc nghẽn hoặc không nhận đủ ACK từ phía nhận. Giai đoạn này có thể là khi TCP đã chuyển sang giai đoạn congestion avoidance.

So sánh với hành vi lý tưởng của TCP

- Trong lý thuyết, giai đoạn slow start sẽ tăng tốc theo cấp số nhân, tức là sau mỗi lần nhận ACK, cửa sổ truyền (congestion window) sẽ tăng gấp đôi. Trong đồ thị thực tế của bạn, quá trình tăng số thứ tự diễn ra khá chậm, không giống như sự tăng theo cấp số nhân, điều này có thể là do điều kiện mạng hoặc cơ chế quản lý luồng của TCP bị ảnh hưởng bởi băng thông hoặc độ trễ. Trong đồ thị thực tế, số thứ tự không thay đổi hoặc tăng rất ít sau giai đoạn slow start, cho thấy có thể có sự mất gói hoặc tắc nghẽn nghiêm trọng hơn trong mạng, dẫn đến TCP không thể tiếp tục tăng tốc.
- Trong lý thuyết, sau khi vượt qua ngưỡng slow start, TCP chuyển sang giai đoạn congestion avoidance, với tốc độ tăng chậm hơn (tuyến tính) để tránh quá tải mạng.

Bình luận

- Điểm khác biệt giữa lý thuyết và thực tế: Trong lý thuyết, TCP sẽ luôn tăng tốc độ truyền trong giai đoạn slow start theo cấp số nhân và chuyển sang congestion avoidance một cách rõ ràng. Tuy nhiên, trong thực tế, mạng có thể gặp nhiều yếu tố như mất gói, độ trễ cao hoặc giới hạn băng thông, khiến cho quá trình này bị chậm lại, dẫn đến việc số thứ tự không tăng nhanh như lý thuyết.

Kết luận: Từ đồ thị của bạn, có thể thấy giai đoạn slow start kết thúc khá sớm (trước 1 giây) và quá trình tránh tắc nghẽn bắt đầu ngay sau đó, nhưng tốc độ truyền dữ liệu giảm mạnh sau khi gặp tắc nghẽn.



14. Trả lời hai câu hỏi trong số các câu hỏi trên cho bản ghi mà bạn đã thu được khi truyền tệp từ máy tính của bạn đến gaia.cs.umass.edu.

Em đã trả lời ở trên.