

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**KHOA AN TOÀN THÔNG TIN**



**Môn học: An toàn mạng**

**Báo Cáo Bài Tập 1**

**Thực hành phân tích HTTP, TCP sử dụng Wireshark**

**Họ và tên:** Trần Thị Thu Phương

**Mã sinh viên:** B21DCAT151

**Nhóm môn học:** 04

**Giảng viên:** Nguyễn Ngọc Điệp

Hà Nội, 8/2024

## Mục lục

<b>I. Wireshark Lab: HTTP .....</b>	<b>1</b>
1. The basic HTTP GET/response interaction .....	1
2. The HTTP CONDITIONAL GET/response interaction .....	3
3. Retrieving Long Documents .....	5
4. HTML Documents with Embedded Objects .....	6
5. HTTP Authentication .....	7
<b>II. Wireshark Lab: TCP.....</b>	<b>9</b>
1. Capturing a bulk TCP transfer from your computer to a remote .....	9
2. A first look at the captured trace .....	10
3. TCP Basics .....	11
4. TCP congestion control in action .....	16

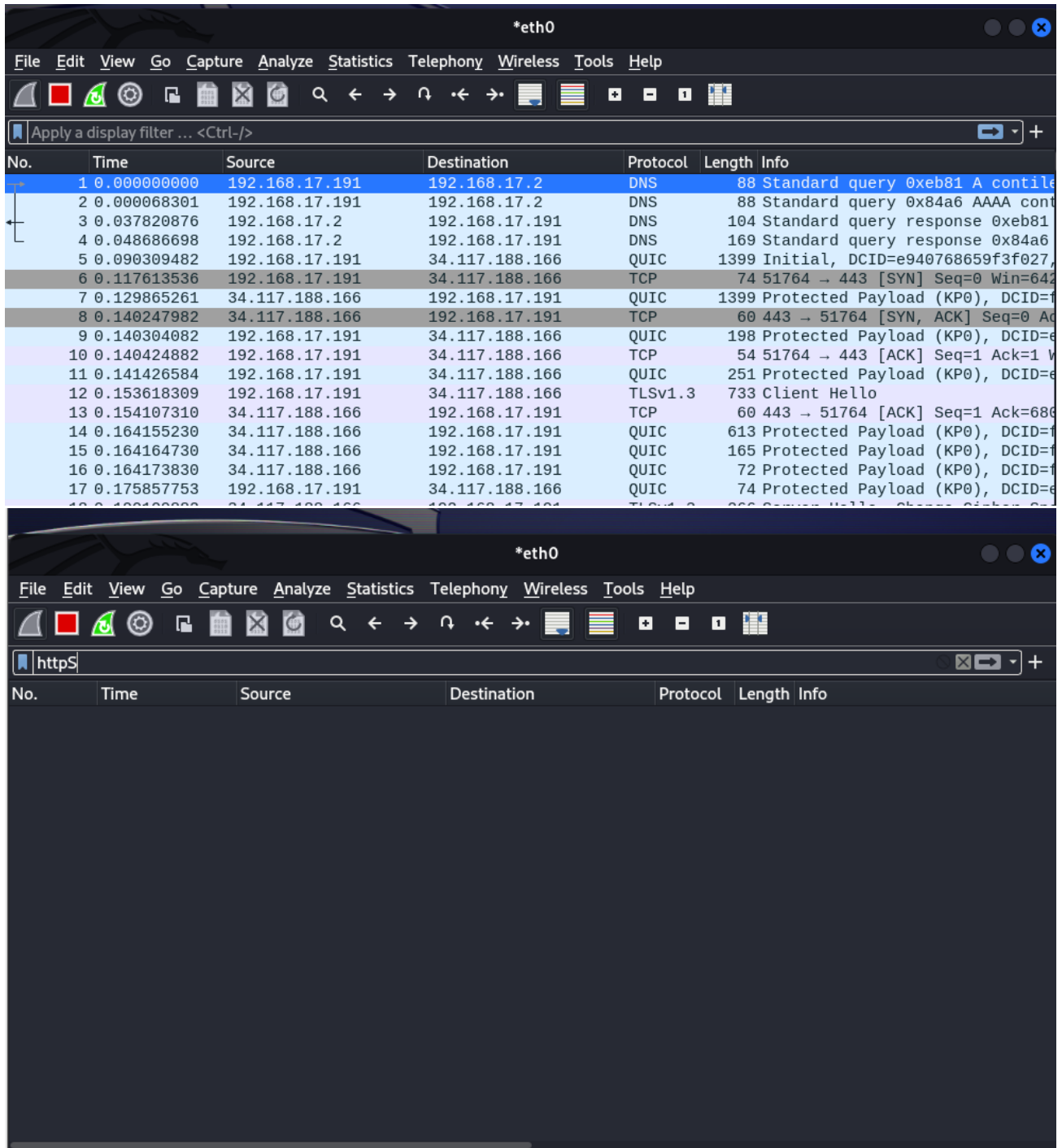
## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

### I. Wireshark Lab: HTTP

#### 1. The basic HTTP GET/response interaction

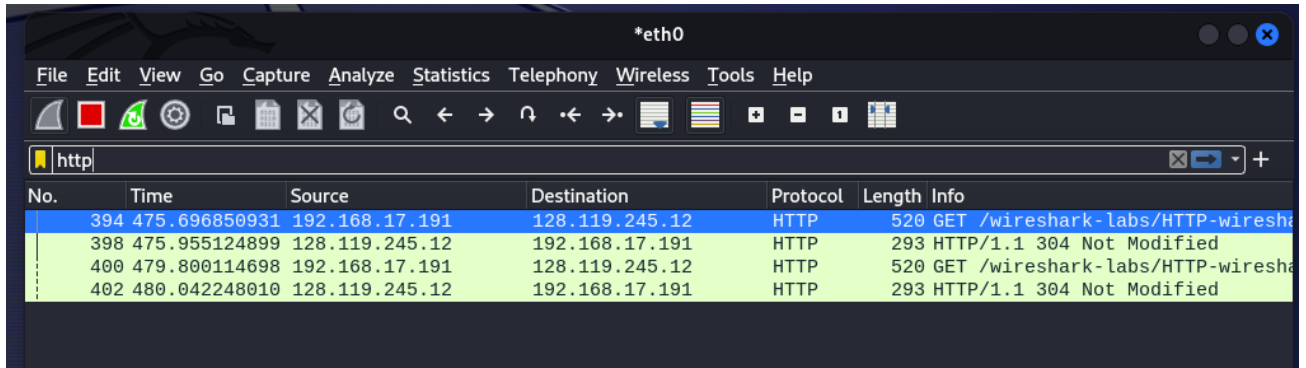
Hãy bắt đầu khám phá HTTP bằng cách tải xuống một tệp HTML đơn giản, tệp này rất ngắn và không chứa các đối tượng nhúng. Làm theo các bước sau đây:

- Khởi động browser
- Khởi động Wireshark và nhập “http” vào thanh filter
- Đợi một hơn một phút (chúng ta sẽ sớm biết lý do tại sao) và sau đó bắt đầu Wireshark packet capture.

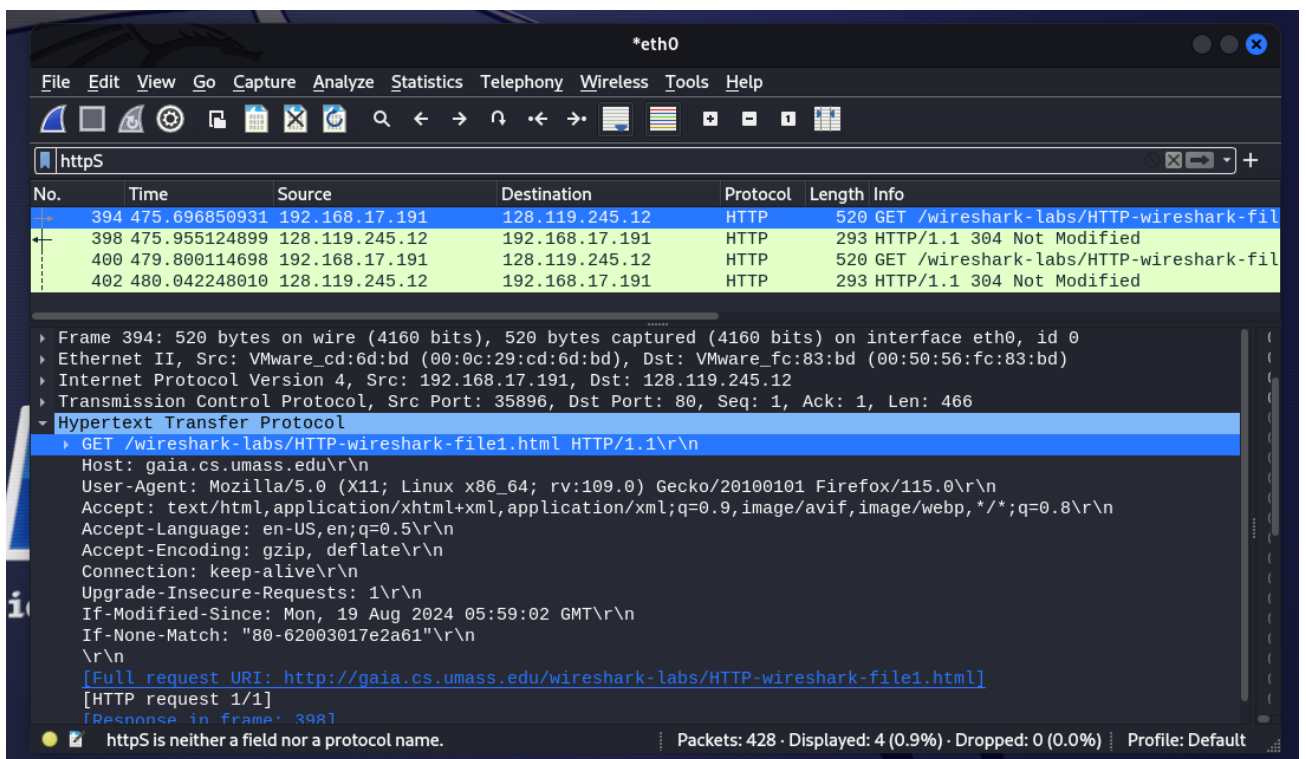


## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

- Nhập <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html> vào trình duyệt



*Kết quả lọc gói tin theo giao thức HTTP*



*Nội dung của gói tin HTTP GET*

- Theo như ví dụ trên, có 2 loại messages HTTP được ghi lại: GET message từ trình duyệt của mình đến web server, response message từ web server đến trình duyệt của mình
- Kết thúc quá trình Wireshark packet capture.

### Trả lời các câu hỏi

1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

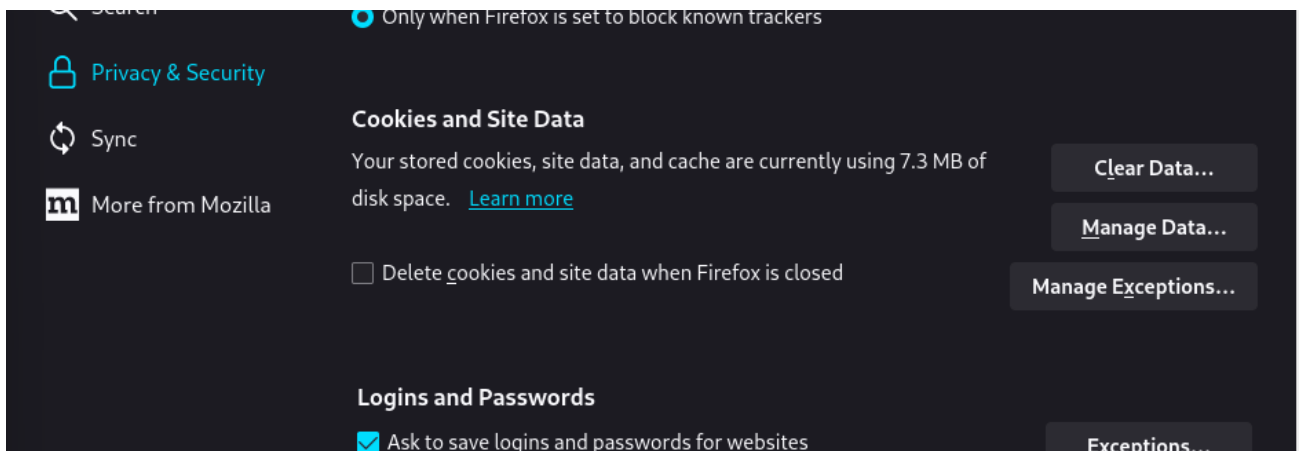
Trình duyệt của em đang chạy HTTP 1.1 và Web server cũng chạy phiên bản 1.1

2. What languages (if any) does your browser indicate that it can accept to the server?  
Ngôn ngữ mà trình duyệt cho biết rằng có thể chấp nhận với máy chủ là: en-US, en; q = 0.5\r\n
3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?  
Địa chỉ IP của máy em là 192.168.17.191, của server là 128.119.245.12
4. What is the status code returned from the server to your browser?  
Status code được trả về từ máy server đến trình duyệt của em là: 200 OK
5. When was the HTML file that you are retrieving last modified at the server?  
Last modified at the server: Mon, 19 Aug 2024 08:41:34 GMT
6. How many bytes of content are being returned to your browser?  
Số lượng byte nội dung được trả về tới trình duyệt của em là: 128
7. By inspecting the raw data in the packet content window, do you see any headers within the data that are not displayed in the packet-listing window? If so, name one.  
Không headers nào hiển thị để thấy

## 2. The HTTP CONDITIONAL GET/response interaction

Các bước thực hiện:

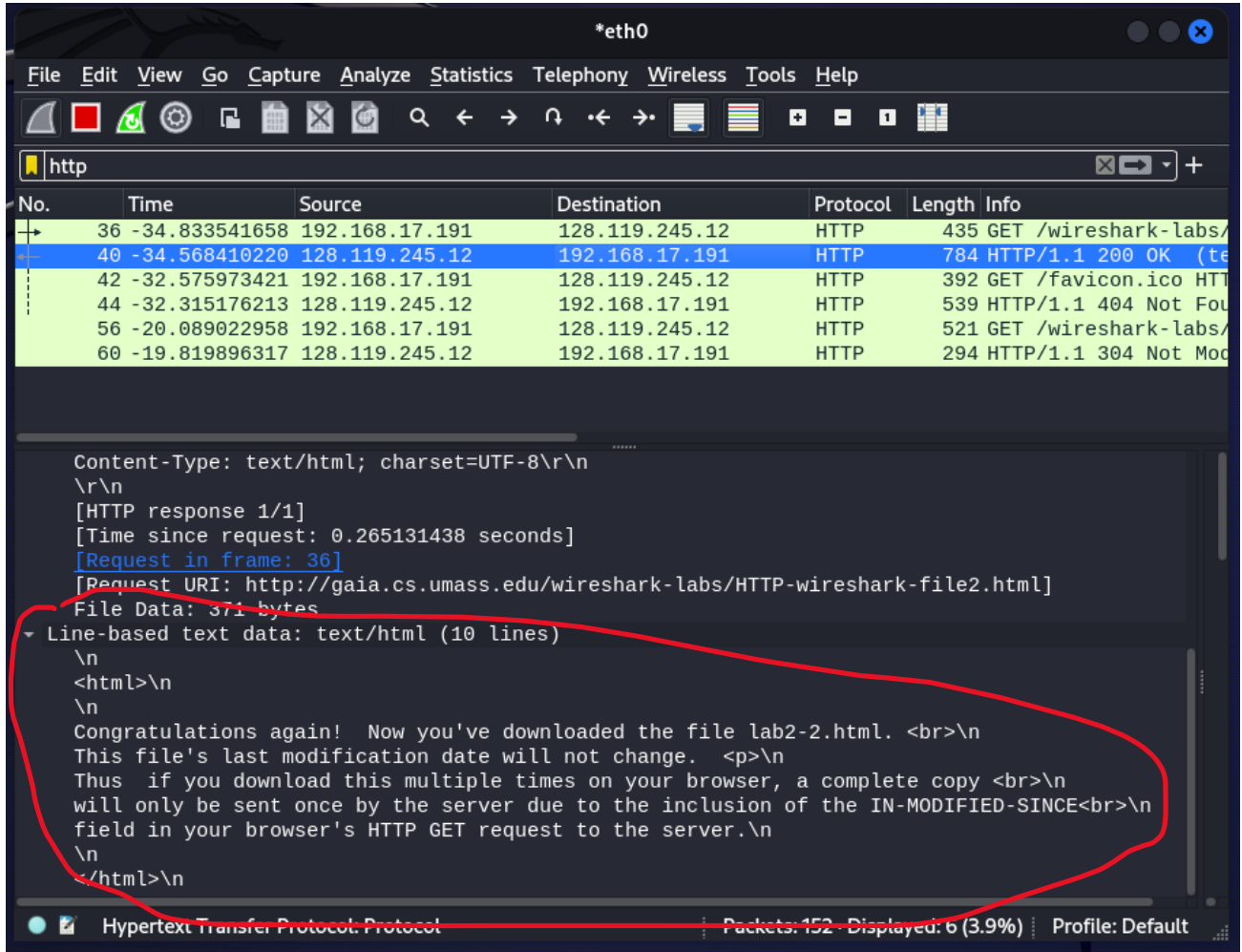
- Khởi động trình duyệt của bạn và đảm bảo rằng bộ nhớ cache trên trình duyệt đã xóa sạch



- Khởi động wireshark

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

- Truy cập trên trình duyệt: <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html>
- Chọn nút làm mới trên trình duyệt
- Dùng Wireshark packet capture, nhập “http” vào thanh lọc để chỉ thấy những gói tin http được ghi lại



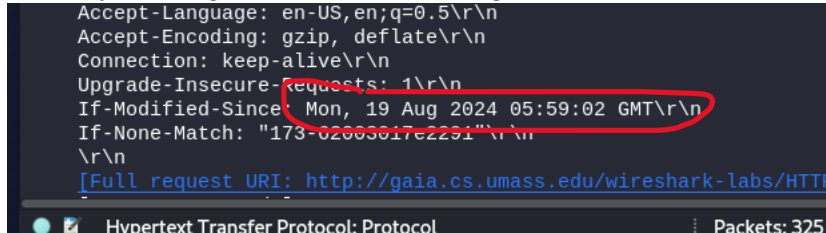
### Trả lời các câu hỏi:

8. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET? Không có dòng nào như thế
9. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell? Có, nội dung rõ ràng nằm trong phần “Line-based text data” như hình trên.

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

- Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE:” line in the HTTP GET? If so, what information follows the “IF-MODIFIED-SINCE:” header?

Có thấy. Thông tin là: Mon, 19 Aug 2024 05:59:02 GMT



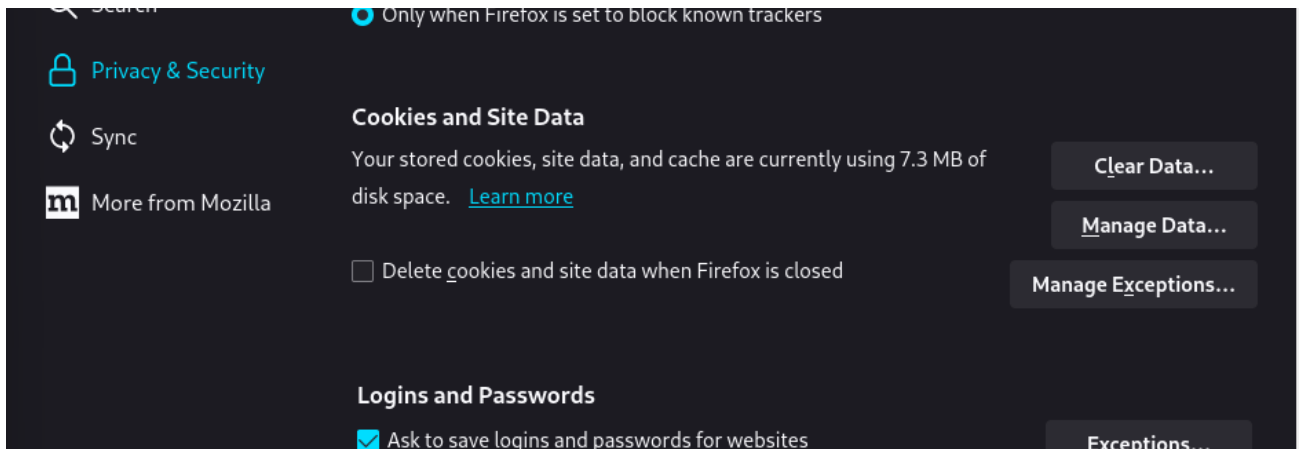
- What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

Status Code: 304 Not Modified, không còn nội dung rõ ràng như lần đầu nữa. Bởi vì trình duyệt sử dụng lại nội dung đã được lưu trong bộ nhớ đệm vì không có thay đổi nào trên máy chủ

### 3. Retrieving Long Documents

Những phần trước đó, chúng ta đã làm việc với tệp HTML đơn giản và ngắn, giờ chúng ta sẽ thử một tệp HTML dài. Các bước thực hiện:

- Khởi động trình duyệt của bạn và đảm bảo rằng bộ nhớ cache trên trình duyệt đã xóa sạch



- Khởi động wireshark
- Truy cập trên trình duyệt: <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html>
- Dừng Wireshark packet capture, nhập “http” vào thanh lọc để chỉ thấy những gói tin http được ghi lại

### Trả lời câu hỏi

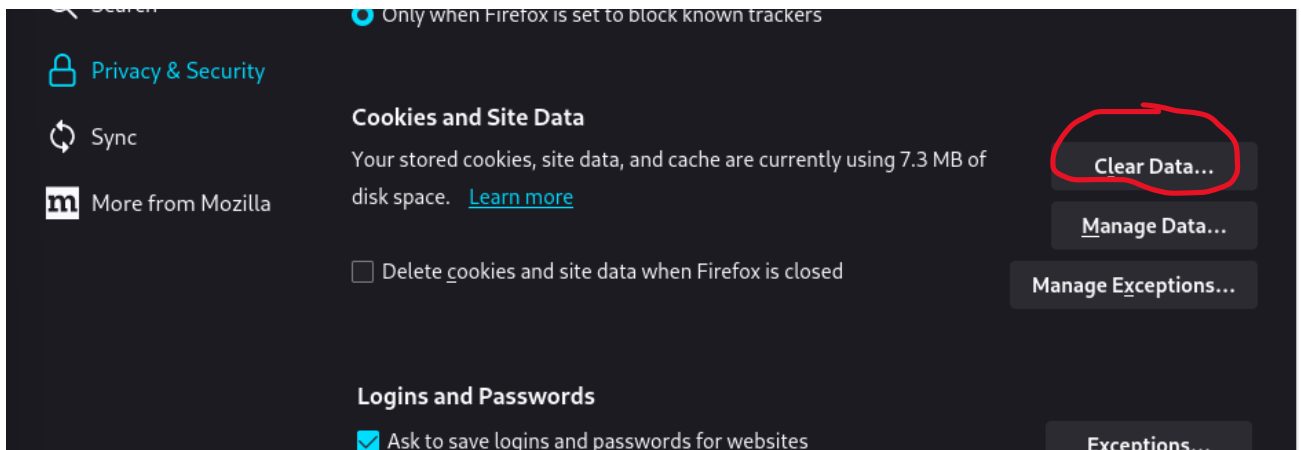
## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

12. How many HTTP GET request messages were sent by your browser?  
2 request messages
13. How many data-containing TCP segments were needed to carry the single HTTP response?  
2 segment TCP
14. What is the status code and phrase associated with the response to the HTTP GET request?  
200 OK
15. Are there any HTTP status lines in the transmitted data associated with a TCP induced “Continuation”?  
Không

### 4. HTML Documents with Embedded Objects

Làm việc với tệp HTML có nhiều đối tượng. các bước thực hiện:

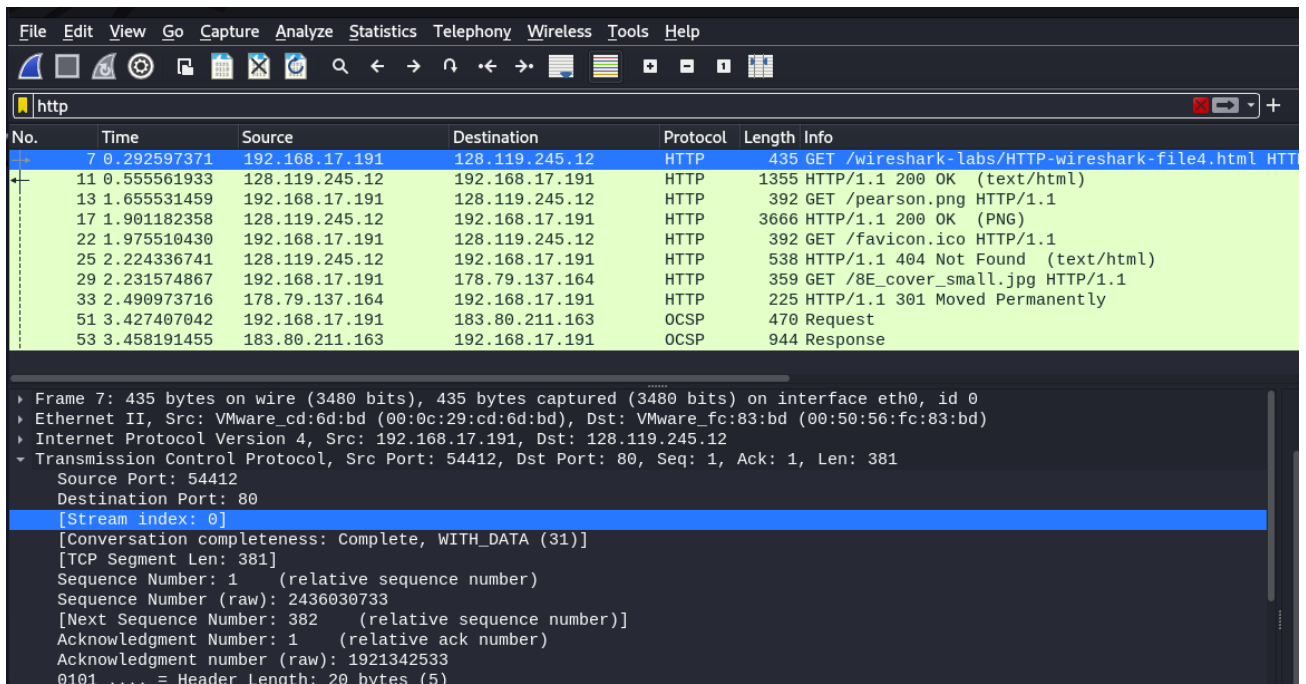
- Khởi động trình duyệt của bạn và đảm bảo rằng bộ nhớ cache trên trình duyệt đã xóa sạch



- Khởi động wireshark
- Truy cập trên trình duyệt: <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html>
- Dừng Wireshark packet capture, nhập “http” vào thanh lọc để chỉ thấy những gói tin http được ghi lại



## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark



The image shows a Wireshark packet capture of an HTTP session. The packet list on the left shows several GET requests and one OCSP response. The packet details pane on the right shows the structure of the selected packet (Frame 7), including Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol fields.

No.	Time	Source	Destination	Protocol	Length	Info
7	0.292597371	192.168.17.191	128.119.245.12	HTTP	435	GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1
11	0.555561933	128.119.245.12	192.168.17.191	HTTP	1355	HTTP/1.1 200 OK (text/html)
13	1.655531459	192.168.17.191	128.119.245.12	HTTP	392	GET /pearson.png HTTP/1.1
17	1.901182358	128.119.245.12	192.168.17.191	HTTP	3666	HTTP/1.1 200 OK (PNG)
22	1.975510430	192.168.17.191	128.119.245.12	HTTP	392	GET /favicon.ico HTTP/1.1
25	2.224336741	128.119.245.12	192.168.17.191	HTTP	538	HTTP/1.1 404 Not Found (text/html)
29	2.231574867	192.168.17.191	178.79.137.164	HTTP	359	GET /8E_cover_small.jpg HTTP/1.1
33	2.490973716	178.79.137.164	192.168.17.191	HTTP	225	HTTP/1.1 301 Moved Permanently
51	3.427407042	192.168.17.191	183.80.211.163	OCSP	470	Request
53	3.458191455	183.80.211.163	192.168.17.191	OCSP	944	Response

Frame 7: 435 bytes on wire (3480 bits), 435 bytes captured (3480 bits) on interface eth0, id 0  
Ethernet II, Src: VMware\_cd:6d:bd (00:0c:29:cd:6d:bd), Dst: VMware\_fc:83:bd (00:50:56:fc:83:bd)  
Internet Protocol Version 4, Src: 192.168.17.191, Dst: 128.119.245.12  
Transmission Control Protocol, Src Port: 54412, Dst Port: 80, Seq: 1, Ack: 1, Len: 381  
Source Port: 54412  
Destination Port: 80  
[Stream index: 0]  
[Conversation completeness: Complete, WITH\_DATA (31)]  
[TCP Segment Len: 381]  
Sequence Number: 1 (relative sequence number)  
Sequence Number (raw): 2436030733  
[Next Sequence Number: 382 (relative sequence number)]  
Acknowledgment Number: 1 (relative ack number)  
Acknowledgment number (raw): 1921342533  
0101 ... = Header Length: 20 bytes (5)

16. How many HTTP GET request messages were sent by your browser? To which Internet addresses were these GET requests sent?

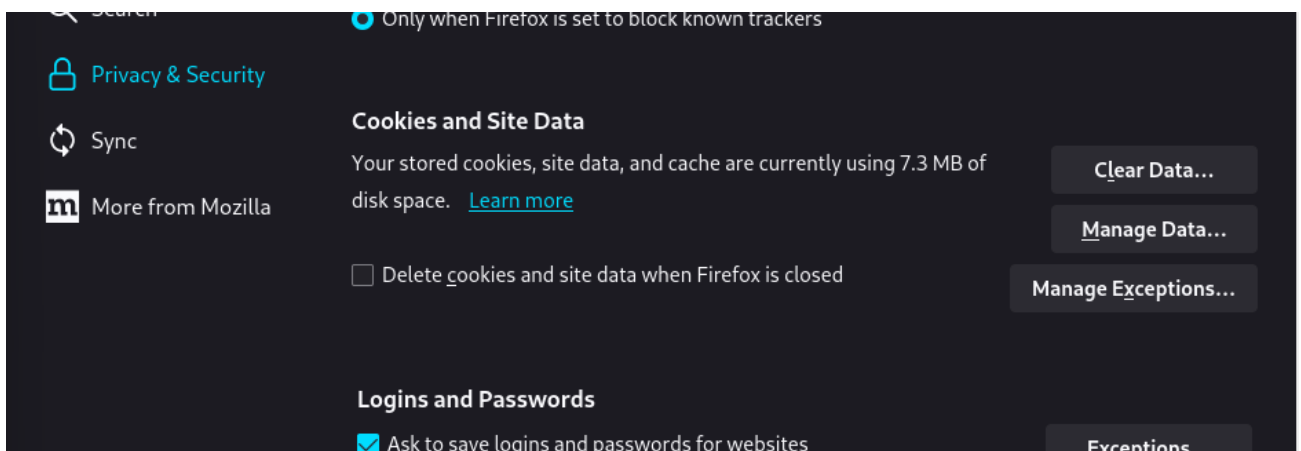
4 HTTP GET, được gửi đến 2 địa chỉ IP: 128.119.245.12, 178.79.137.146

17. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two web sites in parallel? Explain.  
Quá trình download là không đồng thời, ảnh 1 tìm được, ảnh 2 không tìm thấy

### 5. HTTP Authentication

Kiểm tra một trang web được bảo vệ bằng mật khẩu. Các bước thực hiện:

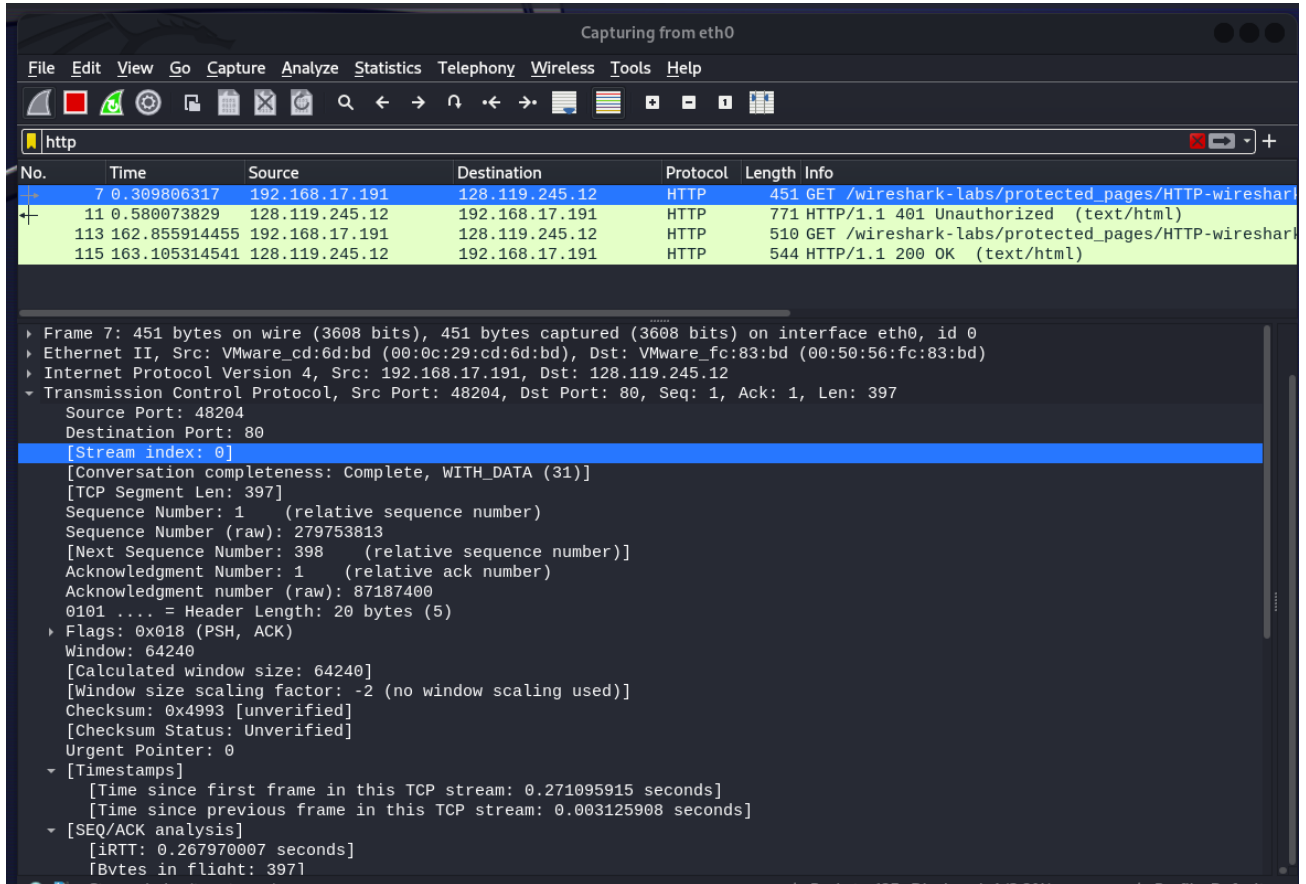
- Khởi động trình duyệt của bạn và đảm bảo rằng bộ nhớ cache trên trình duyệt đã xóa sạch



- Khởi động wireshark

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

- Truy cập trên trình duyệt: <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file5.html> và đăng nhập trang web với username = wireshark-students và password = network.
- Dùng Wireshark packet capture, nhập “http” vào thanh lọc để chỉ thấy những gói tin http được ghi lại



18. What is the server's response (status code and phrase) in response to the initial HTTP GET message from your browser?  
401 Unauthorized
19. When your browser's sends the HTTP GET message for the second time, what new field is included in the HTTP GET message?  
Trường mới là Authorization như hình bên dưới

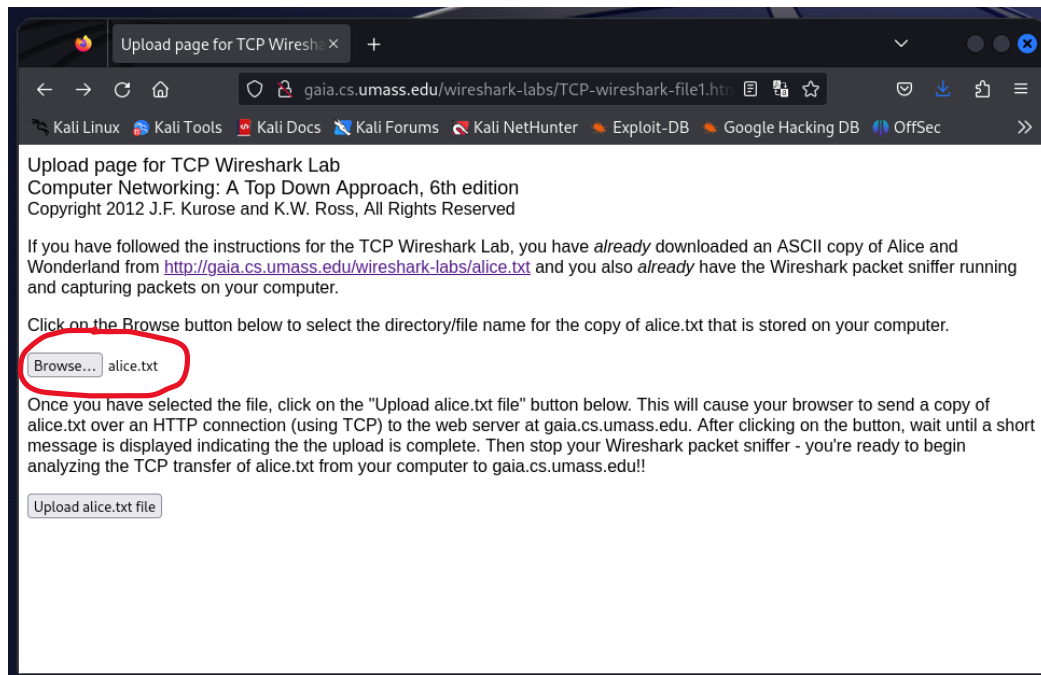
```
Upgrade-Insecure-Requests: 1\r\n
Authorization: Basic d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcm5z\r\n
\r\n
[Full request URI: http://gaia.cs.umass.edu/wirechark-labs/protected
```

## II. Wireshark Lab: TCP

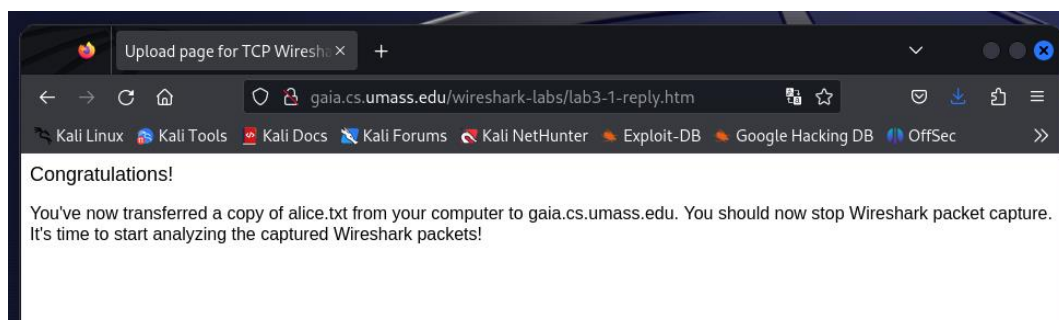
### 1. Capturing a bulk TCP transfer from your computer to a remote

Các bước thực hiện:

- Start up your web browser.
- Go the <http://gaia.cs.umass.edu/wireshark-labs/alice.txt> and retrieve an ASCII copy of Alice in Wonderland. Store this file somewhere on your computer.
- Next go to <http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html>.
- Use the Browse button in this form to enter the name of the file (full path name) on your computer containing Alice in Wonderland (or do so manually). Don't yet press the "Upload alice.txt file" button.



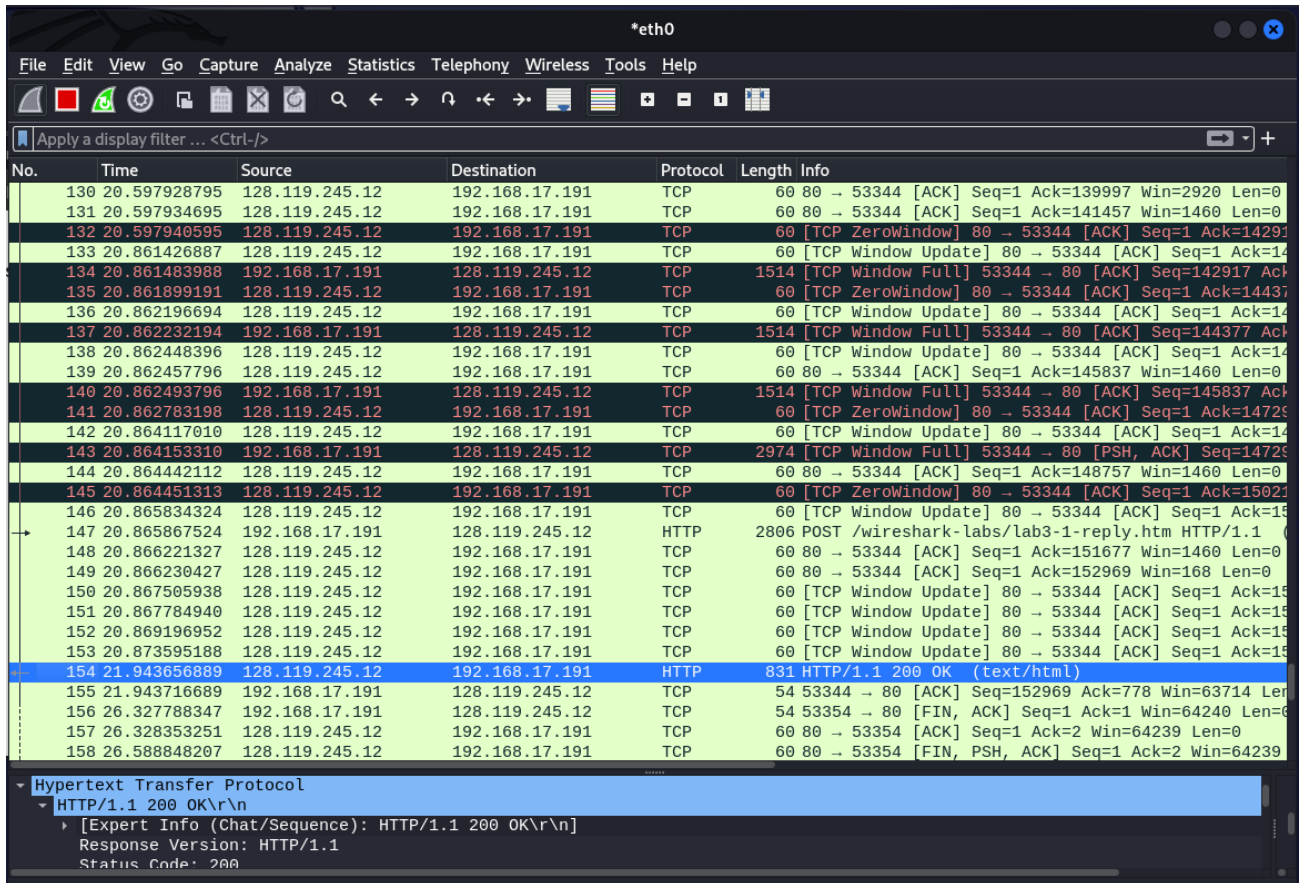
- Now start up Wireshark and begin packet capture (Capture->Options) and then press OK on the Wireshark Packet Capture Options screen (we'll not need to select any options here).
- Returning to your browser, press the "Upload alice.txt file" button to upload the file to the [gaia.cs.umass.edu](http://gaia.cs.umass.edu) server. Once the file has been uploaded, a short congratulations message will be displayed in your browser window.



## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

- Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.

Kết quả như hình bên dưới



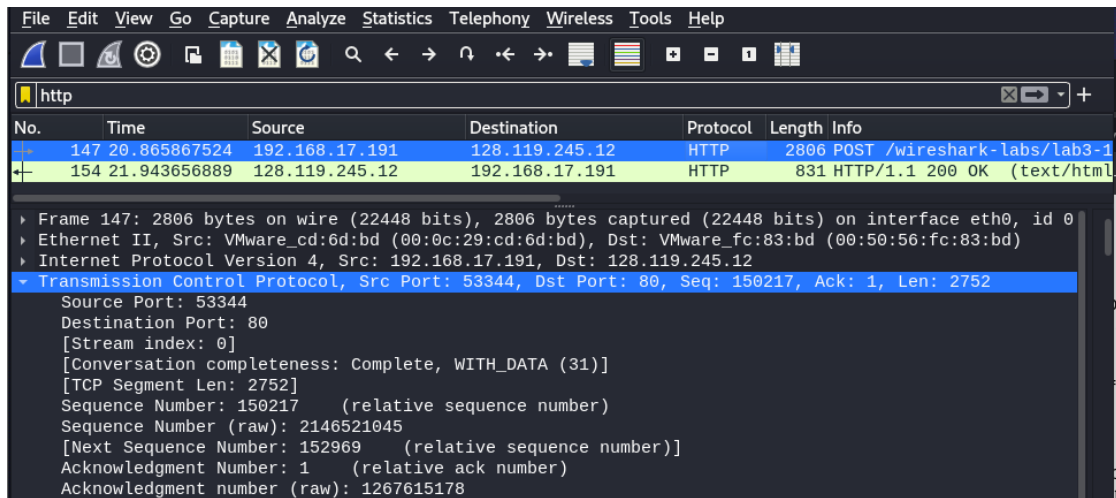
## 2. A first look at the captured trace

Mở tệp tcp.ethereal-trace-1 trong <http://gaia.cs.umass.edu/wireshark-labs/wireshark-trace.zip> và trả lời câu hỏi:

1. What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window" (refer to Figure 2 in the "Getting Started with Wireshark" Lab if you're uncertain about the Wireshark windows).

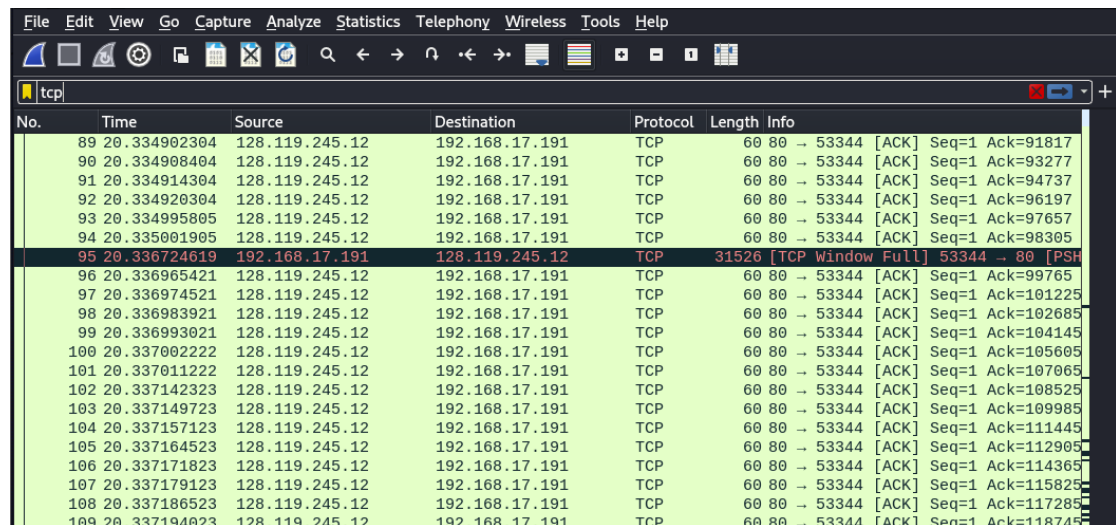
Địa chỉ IP: 192.168.17.191, port 53344

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark



2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

Địa chỉ IP của gaia.cs.umass.edu: 128.119.245.12, cổng gửi và nhận TCP segment cho kết nối này là 80



3. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?

Địa chỉ IP của máy client truyền file là: 192.168.17.191, port 53344

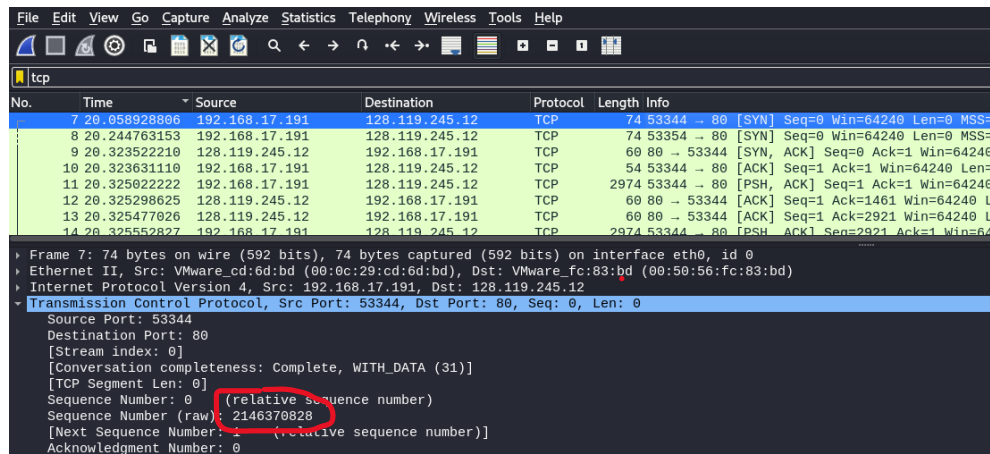
### 3. TCP Basics

#### Trả lời câu hỏi:

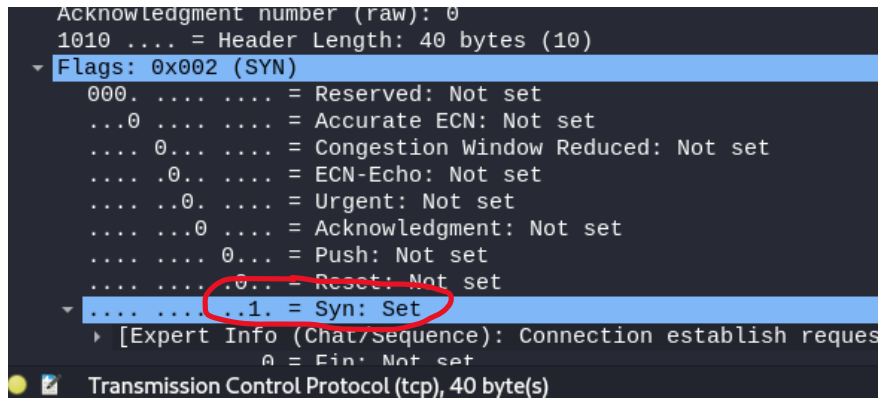
4. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?
- Sequence number của gói tin TCP SYN được sử dụng để khởi tạo kết nối TCP: 2146370828



## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark



- Và điều này xác định phân khúc SYN:



5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the ACKnowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?
  - Sequence number của gói tin SYNACK được gửi từ server tới client để phản hồi gói SYN: 1267615177
  - ACKnowledgement = 2146370829

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

The image shows a Wireshark packet capture. The packet list at the top shows several TCP segments. The selected packet (packet 9) is a TCP segment from 192.168.17.191 to 128.119.245.12, port 80 to 53344. The details pane shows the following information:

- Source Port: 80
- Destination Port: 53344
- [Stream index: 0]
- [Conversation completeness: Complete, WITH\_DATA (31)]
- [TCP Segment Len: 0]
- Sequence Number: 0 (relative sequence number)
- Sequence Number (raw): 1267615177
- [Next Sequence Number: 1 (relative sequence number)]
- Acknowledgment Number: 1 (relative ack number)
- Acknowledgment number (raw): 2146370829
- 0110 .... = Header Length: 24 bytes (0)
- Flags: 0x012 (SYN, ACK)
- Window: 64240
- [Calculated window size: 64240]
- Checksum: 0x6e35 [unverified]
- [Checksum Status: Unverified]
- Urgent Pointer: 0
- Options: (4 bytes), Maximum segment size

- Cách mà server xác định giá trị sequence number của ACK trong SYN/ACK = sequence number của ACK tiếp theo ngay sau đó
- Cách xác định gói đó là gói tin SYN/ACK là cờ được đặt SYN=1, ACK=1

The image shows a close-up of the TCP flags field in the details pane. The flags are: .0.. = ECN-Echo: Not set, .0.. = Urgent: Not set, .1. = Acknowledgment: Set, .0.. = Push: Not set, .0.. = Reset: Not set, .1. = Syn: Set, .0.. = Fin: Not set. The 'Ack: Set' and 'Syn: Set' flags are circled in red.

- What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.
- Sequence number = 2146370829

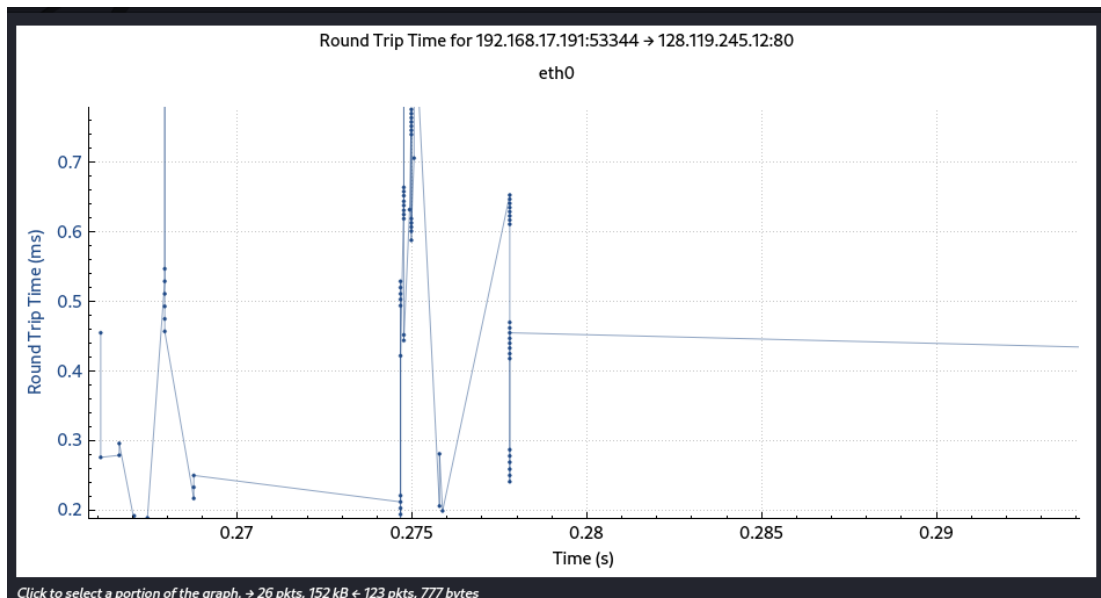
The image shows a Wireshark packet capture. The packet list at the top shows several TCP segments. The selected packet (packet 12) is a TCP segment from 128.119.245.12 to 192.168.17.191, port 53344 to 80. The details pane shows the following information:

- [TCP Segment Len: 2920]
- Sequence Number: 1 (relative sequence number)
- Sequence Number (raw): 2146370829
- [Next Sequence Number: 2920 (relative sequence number)]
- Acknowledgment Number: 1 (relative ack number)
- Acknowledgment number (raw): 1267615178
- 0101 .... = Header Length: 20 bytes (5)
- Flags: 0x018 (PSH, ACK)
- Window: 64240
- [Calculated window size: 64240]
- [Window size scaling factor: -2 (no window scaling used)]
- Checksum: 0x536e [unverified]
- [Checksum Status: Unverified]
- Urgent Pointer: 0
- Timestamps
- [SEQ/ACK analysis]
- [RTT: 0.264702304 seconds]
- [Bytes in flight: 2920]
- [Bytes sent since last PSH flag: 2920]
- TCP payload (2920 bytes)
- Reassembled PDU in frame: 147
- TCP segment data (2920 bytes)

The packet content field at the bottom shows the raw data of the segment, which includes the HTTP POST command.

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? At what time was each segment sent? When was the ACK for each segment received? Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the EstimatedRTT value (see page 249 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 249 for all subsequent segments. Note: Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the “listing of captured packets” window that is being sent from the client to the `gaia.cs.umass.edu` server. Then select: Statistics->TCP Stream Graph >Round Trip Time Graph.



8. What is the length of each of the first six TCP segments?  
Độ dài của 6 TCP segments đầu tiên đều bằng 2920 bytes



## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

tcp

No.	Time	Source	Destination	Protocol	Length	Info
13	20.325477026	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=2921 Win=64240 Len=0
14	20.325552827	192.168.17.191	128.119.245.12	TCP	2974	53344 → 80 [PSH, ACK] Seq=2921 Ack=1 Win=64240 Len=2920
15	20.325831429	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=4381 Win=64240 Len=0
16	20.325848129	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=5841 Win=64240 Len=0
17	20.325980630	192.168.17.191	128.119.245.12	TCP	2974	53344 → 80 [PSH, ACK] Seq=5841 Ack=1 Win=64240 Len=2920
18	20.326131831	192.168.17.191	128.119.245.12	TCP	2974	53344 → 80 [PSH, ACK] Seq=8761 Ack=1 Win=64240 Len=2920
19	20.326165232	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=7301 Win=64240 Len=0
20	20.326172831	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=8761 Win=64240 Len=0
21	20.326278633	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=10221 Win=64240 Len=0
22	20.326286133	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=11681 Win=64240 Len=0
23	20.326290633	192.168.17.191	128.119.245.12	TCP	2974	53344 → 80 [PSH, ACK] Seq=11681 Ack=1 Win=64240 Len=2920

[Conversation completeness: Complete, WITH\_DATA (31)]

[TCP Segment Len: 2920]

Sequence Number: 2921 (relative sequence number)

Sequence Number (raw): 2146373749

[Next Sequence Number: 5841 (relative sequence number)]

Acknowledgment Number: 1 (relative ack number)

Acknowledgment number (raw): 1267615178

0101 ... = Header Length: 20 bytes (5)

Flags: 0x018 (PSH, ACK)

Window: 64240

[Calculated window size: 64240]

[Window size scaling factor: -2 (no window scaling used)]

Checksum: 0x536e [unverified]

[Checksum Status: Unverified]

Urgent Pointer: 0

[Timestamps]

[SEQ/ACK analysis]

[RTT: 0.264702304 seconds]

[Bytes in flight: 2920]

[Bytes sent since last PSH flag: 2920]

TCP payload (2920 bytes)

[Reassembled PDU in frame: 147]

TCP segment data (2920 bytes)

A data segment used in reassembly of a lower-level protocol (tcp.segment\_data), 2920 byte(s)

9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?
  - Lượng không gian bộ đệm khả dụng tối thiểu được quảng cáo ở mức nhận được là 812 byte.

107	20.337179123	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=113825 Win=13952 Len=0
108	20.337186523	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=117285 Win=12492 Len=0
109	20.337194023	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=118745 Win=11032 Len=0
110	20.337335225	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=120205 Win=9572 Len=0
111	20.337341325	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=121665 Win=8112 Len=0
112	20.337347325	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=123125 Win=6652 Len=0
113	20.337353325	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=124585 Win=5192 Len=0
114	20.337359425	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=126045 Win=3732 Len=0
115	20.337365525	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=127505 Win=2272 Len=0
116	20.337371525	128.119.245.12	192.168.17.191	TCP	60	80 → 53344 [ACK] Seq=1 Ack=128965 Win=812 Len=0
117	20.337377625	128.119.245.12	192.168.17.191	TCP	60	[TCP ZeroWindow] 80 → 53344 [ACK] Seq=1 Ack=129777 Win=0
118	20.503893312	128.119.245.12	192.168.17.191	TCP	60	80 → 53354 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=
119	20.504022413	192.168.17.191	128.119.245.12	TCP	54	53354 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
120	20.596397282	128.119.245.12	192.168.17.191	TCP	60	[TCP Window Update] 80 → 53344 [ACK] Seq=1 Ack=129777
121	20.596450682	192.168.17.191	128.119.245.12	TCP	1514	[TCP Window Full] 53344 → 80 [ACK] Seq=129777 Ack=1 W
122	20.596580683	128.119.245.12	192.168.17.191	TCP	60	[TCP ZeroWindow] 80 → 53344 [ACK] Seq=1 Ack=131237 Wi

10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
  - Không, không có phân đoạn nào được truyền lại trong tệp theo dõi. Điều này có thể giải thích là do không tìm thấy các gói có cùng số thứ tự tại các thời điểm khác nhau.
11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 257 in the text).

## Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

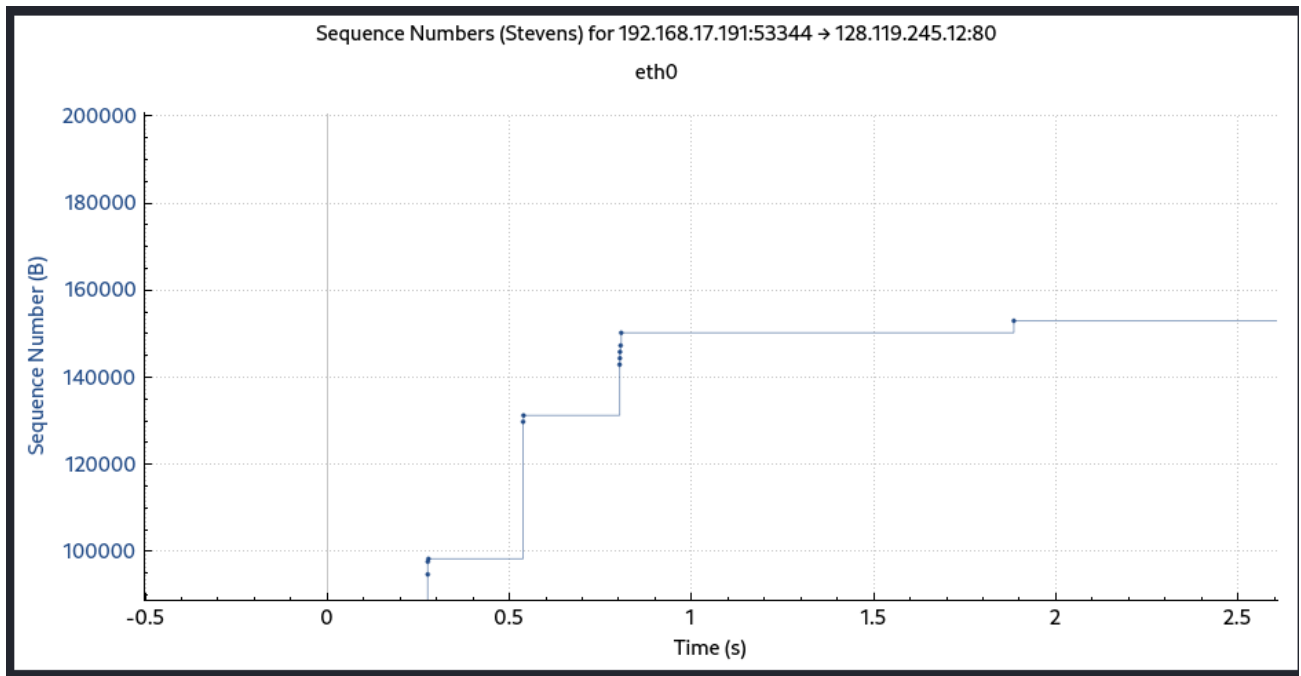
- Người nhận thường thừa nhận 1460 byte trong một ack. Nếu dữ liệu được nhận đôi thì phân khúc đó đang tiếp nhận mọi phân đoạn khác

24	20.326532135	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=13141 Win=64
25	20.326548935	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=13141 Win=64
26	20.326854237	192.168.17.191	128.119.245.12	TCP	14654 53344 → 80	[PSH, ACK] Seq=14601 Ack=1 Win=64
27	20.327311141	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=16061 Win=64
28	20.327329341	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=17521 Win=64
29	20.327347242	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=18981 Win=64
30	20.327365242	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=20441 Win=64
31	20.327383242	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=21901 Win=64
32	20.327401042	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=23361 Win=64
33	20.327684744	192.168.17.191	128.119.245.12	TCP	3622 53344 → 80	[PSH, ACK] Seq=29201 Ack=1 Win=64
34	20.327834945	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=24821 Win=64
35	20.327851646	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=26281 Win=64
36	20.327868046	128.119.245.12	192.168.17.191	TCP	60 80 → 53344	[ACK] Seq=1 Ack=27741 Win=64

Frame 25: 60 bytes on wire (480 bits), 60 bytes captured (480) on interface eth0  
Ethernet II, Src: VMware\_fc:83:bd (00:50:56:fc:83:bd), Dst: 192.168.17.191  
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.17.191

12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.
  - Tập được 152969 byte mà theo tổng thời gian 6.889665575 giây và thông lượng trung bình là 22196,43 byte mỗi giây.

### 4. TCP congestion control in action



### Trả lời câu hỏi

13. Use the Time-Sequence-Graph(Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify where TCP's slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behavior of TCP that we've studied in the text.

Bài tập 1: Thực hành phân tích HTTP, TCP sử dụng Wireshark

- TCP đã tắc nghẽn ở một số đoạn nhưng khoảng thời gian khá ngắn
14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu