**A blue and orange background with dots and lines

Description automatically generated**

GVHD: **Phan Trung Phát**

Lớp: **IT005.O25.2**

|  |  |
| --- | --- |
| **Họ và tên** | **MSSV** |
| Vương Thành Đạt | 23520281 |

2

**Sniffing HTTP Traffic with Wireshark**

**ĐÁNH GIÁ KHÁC (\*):**

|  |  |
| --- | --- |
| **Nội dung** | **Kết quả** |
| Tổng thời gian thực hiện bài thực hành trung bình (1) | 7 giờ |
| Link Video thực hiện (2)  *(nếu có)* |  |
| Ý kiến (3) *(nếu có)*  + Khó khăn  + Đề xuất … |  |
| Điểm tự đánh giá (4) | 9/10 |
| (\*): phần (1) và (4) bắt buộc thực hiện. | |

# Câu hỏi 1: Trình duyệt đang sử dụng phiên bản HTTP 1.0 hay 1.1? Phiên bản HTTP server đang sử dụng là bao nhiêu?

## Minh chứng:

Nội dung.

A screenshot of a computer

Description automatically generated

## Giải thích:

Nội dung.

* Gói tin 29, gói tin này gửi request từ Client đến Server thông qua trình duyệt hiển thị thông tin 413 GET /23520075.html HTTP/1.1. Nên trình duyệt sử dụng phiên bản HTTP 1.1.
* Gói tin 30, gói tin này gửi response từ Server đến Client hiển thị thông tin 660 HTTP/1.1 200 OK(text/html). Nên phiên bản HTTP server đang sử dụng cũng là HTTP 1.1.
* **HTTP/1.1** (Hypertext Transfer Protocol phiên bản 1.1) là một giao thức ứng dụng dùng cho các hệ thống thông tin siêu văn bản phân tán và hợp tác. Nó cho phép xây dựng hệ thống độc lập với dữ liệu đang được truyền, nhờ vào việc định dạng và thương lượng về biểu diễn dữ liệu. Có các tính năng và cải tiến về: lưu trữ Cache, kết nối liên tục, pipelining,…

# Câu hỏi 2: Địa chỉ IP của máy tính bạn là bao nhiêu? Của web server là bao nhiêu?

## Minh chứng:

Nội dung.

A screenshot of a computer

Description automatically generated

## Giải thích:

Nội dung.

Theo như WireShark:

* Địa chỉ IP trên máy tính: 172.20.10.9.
* Địa chỉ IP web server: 172.20.10.13.
* Vì điểm đầu là Client đến Server từ gói HTTP GET đầu tiên thực hiện lệnh request. Nên IP từ 172.20.9 đến 172.20.10.13.

# Câu hỏi 3: Mã trạng thái (status code) trả về từ server là gì?

## Minh chứng:

Nội dung.

A screenshot of a computer

Description automatically generated

## Giải thích:

Nội dung.

+ Có 3 mã trạng thái trong protocol HTTP, nhận biết thông qua điểm IP từ Server về Client:

* Gói tin số 30, status code: 200 OK, là mã phản hồi trạng thái từ máy chủ cho các yêu cầu HTTP thành công từ máy khách (trình duyệt). Đối với một trang web, nó cho biết mã HTML của nó có thể được tải thành công.
* Gói tin số 67, status code: 404 Not Found, là lỗi mã trạng thái HTTP cho biết trang cụ thể mà bạn đang cố truy cập không tồn tại. Thông thường, điều này xảy ra khi miền trỏ đến địa chỉ DNS không chính xác, tệp hoặc thư mục có quyền được định cấu hình sai hoặc hệ thống chạy tệp .htaccess bị hỏng.
* Gói tin số 608, status code: 304 Not Found, là biểu thị không được sửa đổi, cho biết máy chủ không tìm thấy thay đổi nào trên trang được yêu cầu kể từ lần truy cập trước của bạn. Sau đó, trình duyệt của bạn sẽ truy xuất phiên bản được lưu trong bộ nhớ cache của trang web trong bộ nhớ cục bộ của bạn. Bằng cách đó, trình duyệt không phải tải xuống cùng một thông tin từ máy chủ của trang web nhiều lần.

# Câu hỏi 4: Server đã trả về cho trình duyệt bao nhiêu bytes nội dung?

## Minh chứng:

Nội dung.

A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

Description automatically generated

## Giải thích:

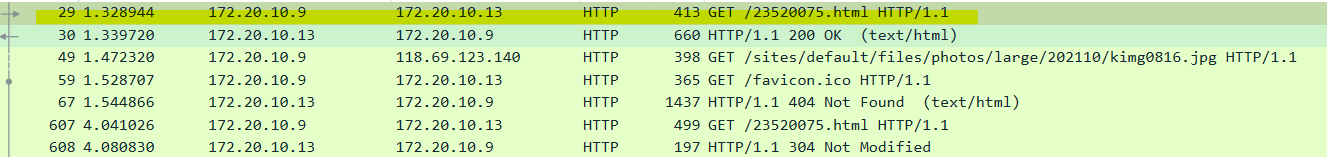
Nội dung.

* Chọn gói HTTP 200 OK, ở phần Packet Details, trong Hypertext Transfer Protocol, ta thấy mục Content-Length và mục File Data đều có 381 bytes. So sánh ở phần Packet Draw, ta thấy nội dung dữ liệu là các đoạn code HTML kèm URL trong phần notepad đã soạn trước đó.

# Câu hỏi 5: Xem xét nội dung của HTTP GET đầu tiên. Bạn có thấy dòng “IF-MODIFIEDSINCE” hay không?

## Minh chứng:

Nội dung.

A computer screen shot of a computer screen

Description automatically generated

## Giải thích:

Nội dung.

* Ở HTTP GET đầu tiên, không tìm thấy “IF - MODIFIEDSINCE”.

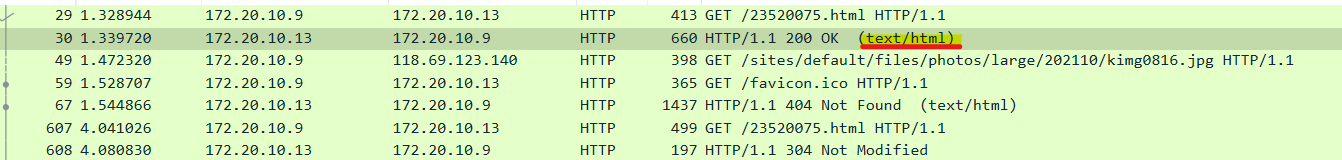
Theo tìm hiểu thì từ khóa “IF - MODIFIEDSINCE”:

* Là một **header field** trong giao thức **HTTP**. Nó được sử dụng để kiểm tra xem một tài nguyên trên máy chủ có thay đổi kể từ thời điểm cuối cùng mà trình duyệt đã yêu cầu nó hay không.
* Khi trình duyệt gửi một yêu cầu **GET** đến máy chủ, nó có thể bao gồm một trường **IF-MODIFIED-SINCE** trong header của yêu cầu. Giá trị của trường này là một **thời gian** (thường là thời gian cuối cùng mà trình duyệt đã nhận được nội dung của tài nguyên từ máy chủ).
* Khi máy chủ nhận được yêu cầu, nó sẽ kiểm tra thời gian cuối cùng mà tài nguyên đã được thay đổi. Nếu thời gian này sau thời gian được chỉ định trong trường **IF-MODIFIED-SINCE**, máy chủ sẽ trả về mã trạng thái **304 Not Modified** mà không gửi lại nội dung của tài nguyên. Trong trường hợp này, trình duyệt sẽ sử dụng phiên bản tải về từ bộ nhớ cache của nó.
* Nếu tài nguyên đã thay đổi kể từ thời điểm cuối cùng, máy chủ sẽ trả về nội dung mới và mã trạng thái **200 OK**
* Tiêu đề HTTP yêu cầu If-Modified-Since làm cho yêu cầu có điều kiện: máy chủ gửi lại tài nguyên được yêu cầu, với trạng thái 200, chỉ khi nó được sửa đổi lần cuối sau ngày nhất định. Nếu tài nguyên chưa được sửa đổi kể từ đó, phản hồi là 304 không có nội dung; tiêu đề phản hồi được sửa đổi lần cuối của yêu cầu trước đó chứa ngày sửa đổi lần cuối. Không giống như If-Unmodified-Since, If-Modified-Since chỉ có thể được sử dụng với GET hoặc HEAD
* Vậy việc xóa Cache trong trình duyệt đã làm mới lại trong việc kiểm tra tài nguyên trên máy đã dùng tại thời điểm đó nên không có xuất hiện.

# Câu hỏi 6: Xem xét nội dung phản hồi từ server. Server có thật sự trả về nội dung của file HTML hay không? Tại sao?

## Minh chứng:

Nội dung.



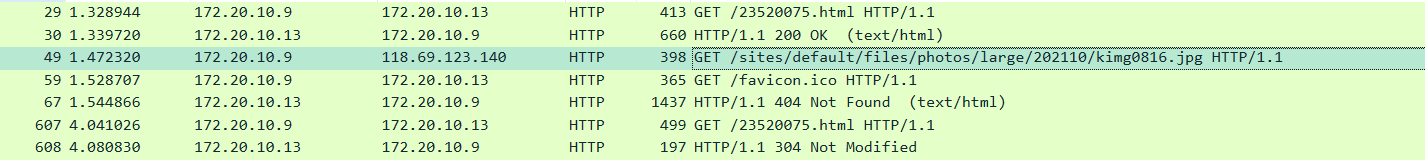
## Giải thích:

Nội dung.

* Dựa vào HTTP 200 OK chứa 381 bytes, đã xóa Cache trước khi sử dụng trình duyệt. Nên khi khởi tạo liên kết, Server không cần phải kiểm tra tài nguyên kể từ lần cuối dùng trình duyệt mà chỉ cần trả về nội dung HTML, và việc không thấy nếu không có trường **IF-MODIFIED-SINCE** cũng là bình thường vì ta không thấy nó trong phản hồi trên.

# Câu hỏi 7: Xem xét nội dung của HTTP GET thứ 2. Bạn có thấy dòng “IF-MODIFIEDSINCE” hay không? Nếu có, giá trị của IF-MODIFIED-SINCE là gì?

## Minh chứng:

Nội dung. A screenshot of a computer

Description automatically generated

## A screenshot of a computer Description automatically generatedA screenshot of a computer Description automatically generatedGiải thích:

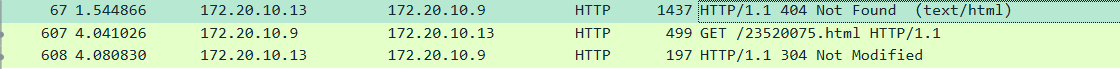
Nội dung.

* Ở gói số 49, không thấy dòng “IF-MODIFIEDSINCE”, theo cảm nhận cá nhân thì lúc này chưa có tải URI trước đó vì đã xóa Cache trình duyệt.
* Ở gói số 59, cũng không thấy dòng “IF-MODIFIEDSINCE”, HTTP GET favicon(viết tắt của “favorite icon” hoặc “shortcut icon”) là một biểu tượng nhỏ xuất hiện bên cạnh địa chỉ website trên thanh địa chỉ của trình duyệt, giúp user dễ nhận diện.
* Ở gói số 607, có xuất hiện “IF-MODIFIEDSINCE”, Thursday, 28 March 2024 08:26:18 GMT\r\n (giờ chuẩn GreenWich- tiêu chuẩn thời gian làm cơ sở cho việc điều phối các múi giờ trên toàn thế giới)
* Gói 607 là HTTP GET thứ 2 đề yêu cầu.

# Câu hỏi 8: Mã trạng thái HTTP được trả về từ server tương ứng với HTTP GET thứ 2 là gì? Ý nghĩa nó là gì? Server có thật sự gửi về nội dung của file hay không? Giải thích.

## Minh chứng:

Nội dung.



## Giải thích:

Nội dung.

* Mã trạng thái HTTP được trả về từ Server ứng với HTTP GET thứ 2 là: 304 Not Modified.
* 304 Not Modified: là mã phản hồi của HTTP cho biết không cần phải truyền lại tài nguyên được yêu cầu. Mã phản hồi này được gửi khi yêu If-None-Match hoặc If-Modified-Since và điều kiện đánh giá sai. Đây là một chuyển hướng ngầm định đến tài nguyên được lưu trong bộ nhớ cache, tài nguyên này sẽ trả về mã phản hồi 200 OK nếu điều kiện đánh giá đúng là một yêu cầu GET hoặc HEAD có điều kiện với tiêu đề. Phản hồi không được chứa nội dung và phải bao gồm các tiêu đề mà đã được gửi trong phản hồi tương đương 200 OK: Cache Control, Content-Location, Date, Etag, Expires và Vary.
* Vậy khi Reset lại website, trình duyệt đã lưu sẵn file ở lần GET file request đầu tiên vào Cache, việc Reset lại là gửi mới GET file request lần hai, 2 phương thức trùng lặp nên trình duyệt sẽ tự động lấy file đầu tiên lưu trong Cache ra và dùng. Server không cần phải gửi lại nữa nên sẽ trả về mã trạng thái 304 như đã giải thích ở trên.

# Câu hỏi 9: Trình duyệt đã gửi bao nhiêu HTTP GET? Đến những địa chỉ IP nào?

## Minh chứng:

Nội dung. 

## Giải thích:

Nội dung.

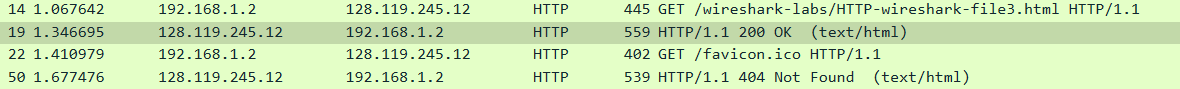
* Trình duyệt gửi HTTP GET đến Server trang web đã tạo và Server URI link ảnh trong website.

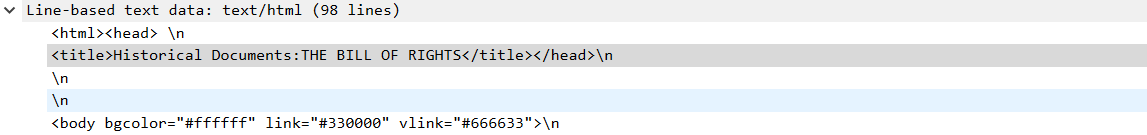
Cụ thể:

* IP Server website: 172.20.10.13. Đây là lần đầu tiên trình duyệt gửi request đến Server tạo liên kết để cấp quyền truy cập website đó.
* IP Server ảnh: 118.69.123.140. Đây là lần đầu tiên trình duyệt yêu cầu đường dẫn URI link ảnh http… để truy cập đến ảnh đó.

# Câu hỏi 10: Trình duyệt đã gửi bao nhiêu HTTP GET? Dòng “THE BILL OF RIGHTS” được chứa trong gói tin phản hồi thứ mấy?

## Minh chứng:

Nội dung. 



## Giải thích:

Nội dung.

* Trình duyệt gửi 2 gói tin HTTP GET ở gói 14 và 22 vì địa chỉ IP xuất phát là của Client.
* Dòng “THE BILL OF RIGHTS” được chứa trong gói tin 19 HTTP 200 OK, trong phần Packet Details mục Line-based text data.

# Câu hỏi 11: Cần bao nhiêu TCP segments để chứa hết HTTP response và nội dung của The Bill of Rights?

## Minh chứng:

Nội dung.

A screenshot of a computer

Description automatically generated

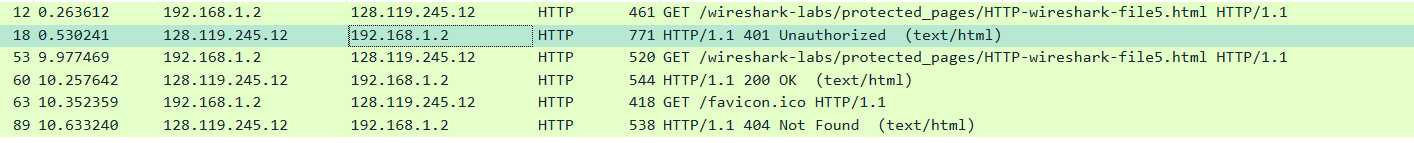
## Giải thích:

Nội dung.

* Cần 4 TCP segments để chứa hết HTTP response và nội dung của The Bill of Rights, thông thường sẽ chia nhỏ các TCP segments có kích thước bằng nhau đến khi đủ thì thôi.

# Câu hỏi 12: Mã trạng thái và ý nghĩa HTTP response tương ứng với HTTP GET đầu tiên là gì?

## Minh chứng:

Nội dung. 

## Giải thích:

Nội dung.

* Mã trạng thái HTTP response tương ứng với HTTP GET đầu tiên là:

401 Unauthorized.

* 401 Unauthorized: cho biết rằng yêu cầu của khách hàng chưa được hoàn thành vì nó thiếu thông tin xác thực hợp lệ cho tài nguyên được yêu cầu. Mã trạng thái này được gửi cùng với tiêu đề phản hồi HTTP WWW-Authenticate chứa thông tin về cách khách hàng có thể yêu cầu lại tài nguyên sau khi nhắc người dùng về thông tin xác thực.

# Câu hỏi 13: Khi trình duyệt gửi HTTP GET lần thứ 2, trường dữ liệu nào mới nào xuất hiện trong HTTP GET? Hãy giải thích ý nghĩa và các vấn đề liên quan của trường mới này.

## Minh chứng:

Nội dung.

A screenshot of a computer

Description automatically generated

## Giải thích:

Nội dung.

* Xuất hiện trường mới là Authorization, có lưu Credentials(thông tin xác thực) của user và password.
* Authorization: yêu cầu Ủy quyền HTTP có thể được sử dụng để cung cấp thông tin xác thực xác thực tác nhân người dùng với máy chủ, cho phép truy cập vào tài nguyên được bảo vệ.

**YÊU CẦU CHUNG**

1. Đánh giá

* Chuẩn bị tốt các yêu cầu đặt ra trong bài thực hành.
* Sinh viên hiểu và tự thực hiện được bài thực hành, trả lời đầy đủ các yêu cầu đặt ra.
* Nộp báo cáo kết quả chi tiết những đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả *(nếu có)*; giải thích cho quan sát *(nếu có)*.
* Sinh viên báo cáo kết quả thực hiện và nộp bài.

1. Báo cáo

* Nộp file .docx. Tập trung vào nội dung, giải thích.
* Nội dung trình bày bằng Font chữ Cambria hoặc Times New Roman (*tuy nhiên, phải chuyển đổi hết báo cáo này sang 1 font chữ thống nhất*) – cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.
* Đặt tên theo định dạng: Mã lớp-LabX\_MSSV1\_MSSV2. (trong đó X là Thứ tự buổi Thực hành).

Ví dụ: IT005.O21.1-Lab01\_25520001\_25520002

* Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
* Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài thực hành.
* Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Bài sao chép, trễ, … sẽ được xử lý tùy mức độ vi phạm.

**HẾT**