

HW2 Report

Computer Network

2013313217 조상연

1. Development Environments

1. Python Version: Python 3.8.1
2. IDE: VS Code
3. OS: Mac OS X 10.15.3

2. Getting Started

- 같은 폴더에 html, image 파일들과 2013313217.py를 넣고, 해당 폴더로 이동 후 아래 명령어 실행
- "listen"이 나오면 정상적으로 실행 중임을 알 수 있다.

```
python 2013313217.py
```

```
# python3 가 기본이 아니라 오류가 난다면  
python3 2013313217.py
```

3. Design

3.1 사용한 모듈

```
import os  
import time  
import threading  
import socket
```

3.2 전체 흐름 및 설계

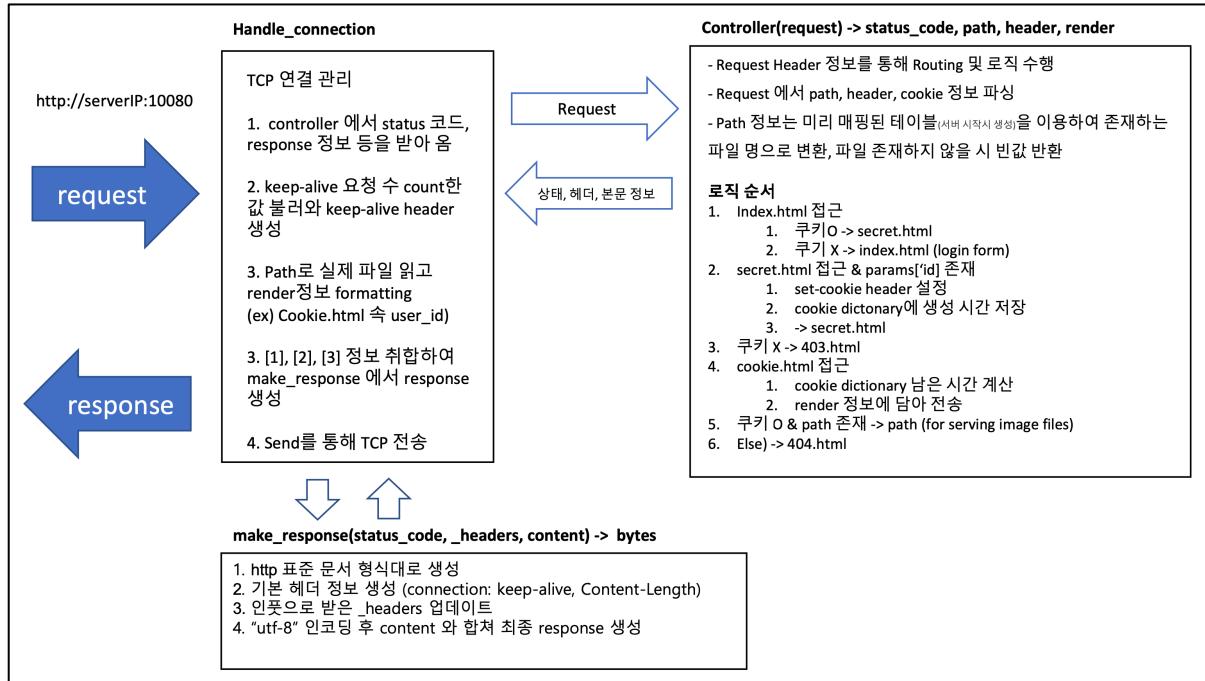


그림 1 전체 흐름도

Socket에서 생성한 connection을 handle_connection에 넘기며 **Thread**를 생성한다. 해당 Thread는 persistent HTTP를 유지하기 위해 keep-alive max_count를 넘기지 않거나 timeout 전이라면 connection을 계속 유지하도록 한다.

Index.html, secret.html, cookie.html, img 파일들, 404.html, 403.html의 문서와 이미지 파일을 준비하였고 이를 같은 폴더에 넣어 경로를 모두 같게 하였다.

4.1 Basic HTTP requests & responses (60 points)

- User requests HTML or image files.

If the HTML file includes embed images, the browser requests them automatically. Therefore, your web server program must handle those subsequent requests.

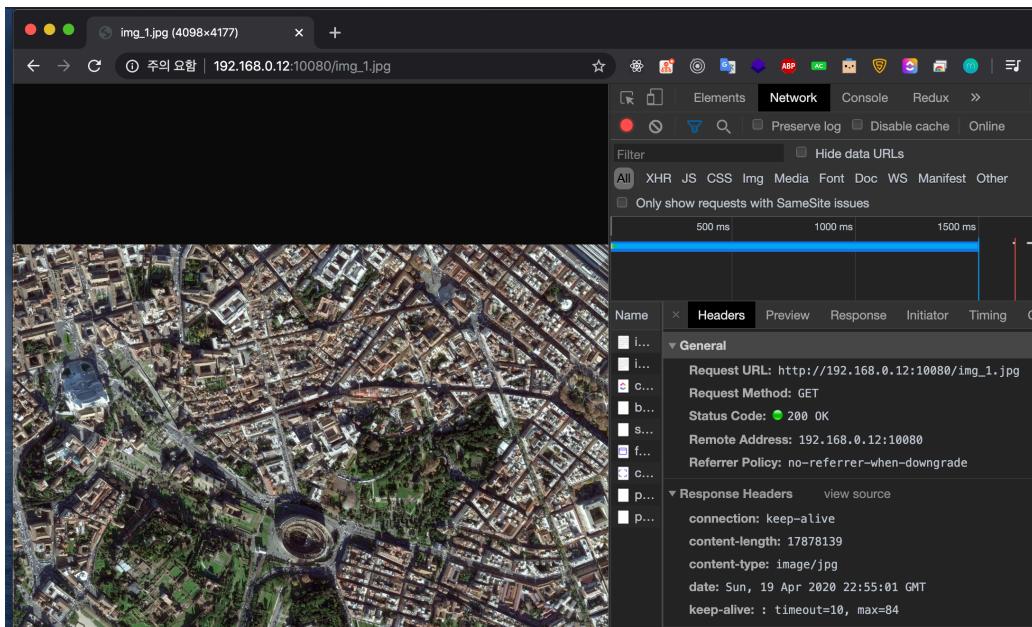


그림 2 이미지 파일 요청 및 응답

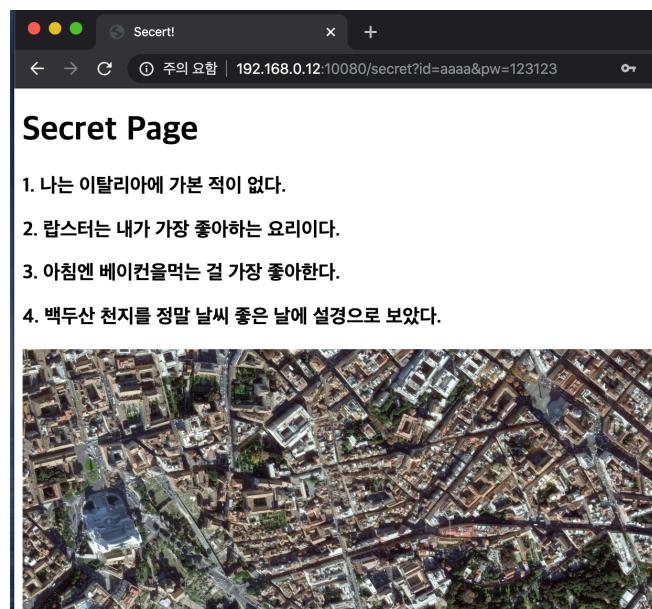


그림 3 HTML 문서 및 Embed 된 사진 모두 정상적으로 응답

- Image file size can be up to O(10) Mbytes.

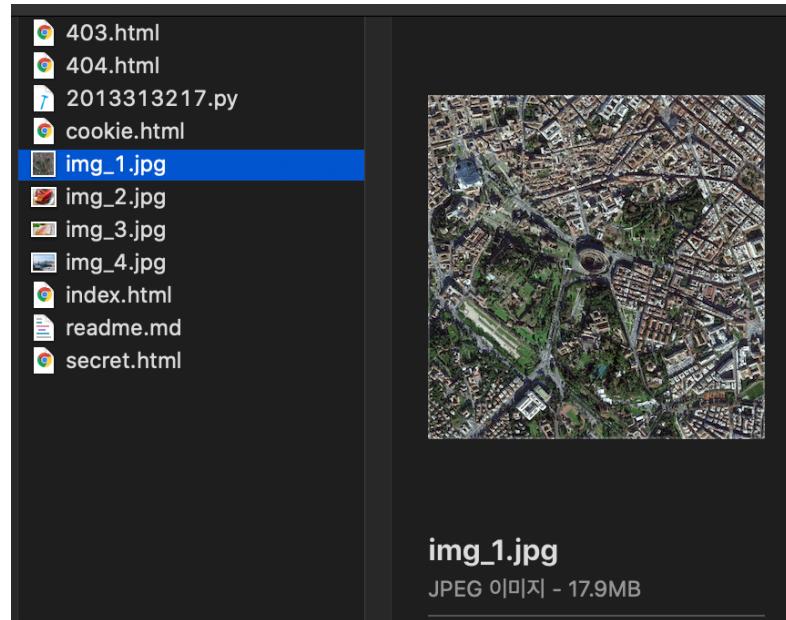


그림 4 이미지 용량 17.9MB

- User can repeatedly access / request any content in Web server.

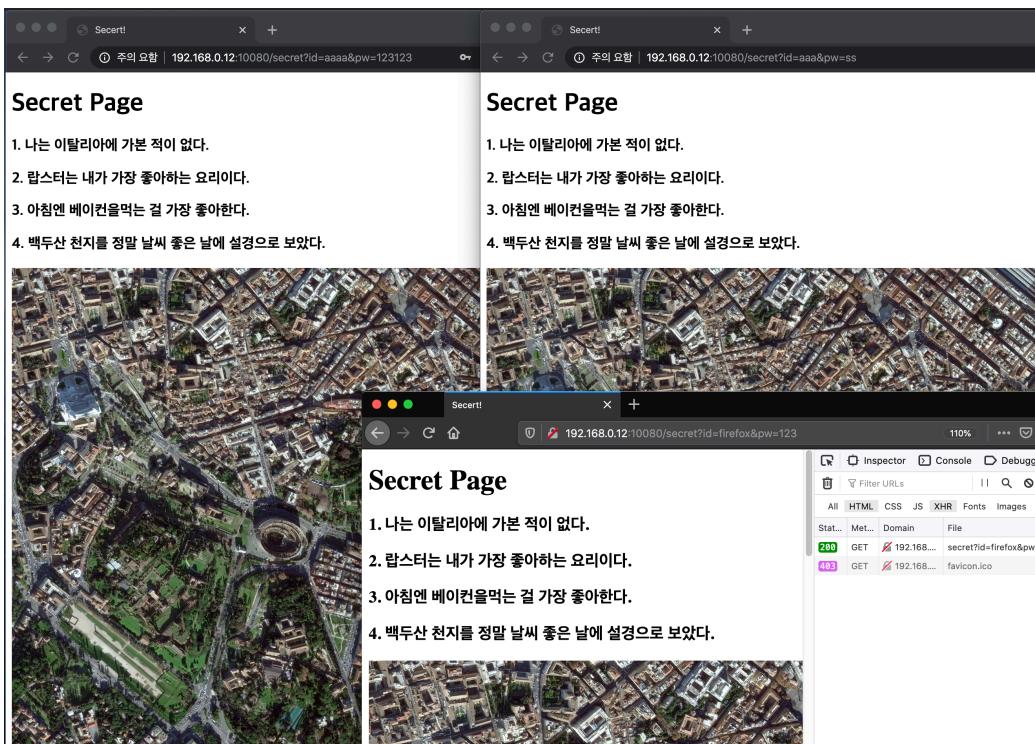


그림 5 여러 브라우저에서 동시에 다발적 접속, 다수의 새로고침 후 스크린샷

- If you request a file that is not present in the web server machine, the web server should return a "404 Not Found" error message.

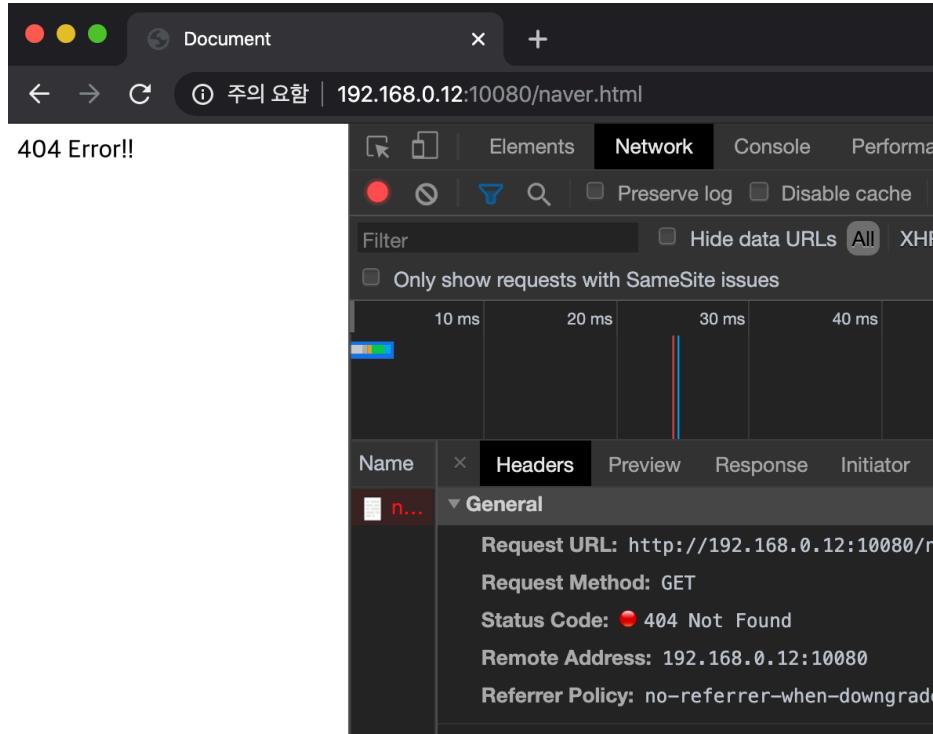


그림 6 404 Error Message 정상적 출력

- The web server program can work in a non-persistent or a persistent HTTP mode.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168...	192.168.0.12	HTTP	573	GET /cookie.html HTTP/1.1
2	0.000026	192.168...	192.168.0.12	TCP	56	10080 → 56054 [ACK] Seq=1 Ack=51
3	0.000290	192.168...	192.168.0.12	HTTP	530	HTTP/1.1 200 OK (text/html)
4	0.000311	192.168...	192.168.0.12	TCP	56	56054 → 10080 [ACK] Seq=518 Ack=530
•	21 2.108399	192.168...	192.168.0.12	HTTP	573	GET /cookie.html HTTP/1.1
•	22 2.108421	192.168...	192.168.0.12	TCP	56	10080 → 56054 [ACK] Seq=475 Ack=530
•	23 2.108667	192.168...	192.168.0.12	HTTP	530	HTTP/1.1 200 OK (text/html)
•	24 2.108688	192.168...	192.168.0.12	TCP	56	56054 → 10080 [ACK] Seq=1035 Ack=530
•	33 2.849656	192.168...	192.168.0.12	HTTP	573	GET /cookie.html HTTP/1.1
•	34 2.849676	192.168...	192.168.0.12	TCP	56	10080 → 56054 [ACK] Seq=949 Ack=530
•	35 2.849925	192.168...	192.168.0.12	HTTP	530	HTTP/1.1 200 OK (text/html)
•	36 2.849940	192.168...	192.168.0.12	TCP	56	56054 → 10080 [ACK] Seq=1552 Ack=530
•	37 3.224270	192.168...	192.168.0.12	HTTP	573	GET /cookie.html HTTP/1.1
•	38 3.224318	192.168...	192.168.0.12	TCP	56	10080 → 56054 [ACK] Seq=1423 Ack=530
•	39 3.224637	192.168...	192.168.0.12	HTTP	530	HTTP/1.1 200 OK (text/html)
•	40 3.224652	192.168...	192.168.0.12	TCP	56	56054 → 10080 [ACK] Seq=2069 Ack=530

```

TCP payload (474 bytes)
▼ Hypertext Transfer Protocol
▶ HTTP/1.1 200 OK\r\n
date: Sun, 19 Apr 2020 23:04:33 GMT\r\n
connection: keep-alive\r\n
▶ content-length: 300\r\n
content-type: text/html; charset=utf-8\r\n
keep-alive:: timeout=10, max=96\r\n
\r\n
[HTTP response 4/8]
[Time since request: 0.000367000 seconds]
[Prev request in frame: 33]
[Prev response in frame: 35]
[Request in frame: 37]
[Next request in frame: 49]
[Next response in frame: 51]
[Request URI: http://192.168.0.12:10080/cookie.html]
File Data: 300 bytes

```

그림 7 HTTP persistent mode

아래 그림에서 connection: keep-alive 와 HTTP response 4/8 을 통해 persistent mode 가 정상적으로 작동함을 알 수 있다. 위 스크린샷에서 최초 handshake 이후 그 이후에 request에 대해선 하지 않는 걸 알 수 있다.

4.2 Log-in functionality (20 points)

- If you add 'Log-in functionality', users must access "http://serverIP:10080" for the first time, and the web server sends "index.html".
- The "index.html" file shows the following input forms

Hello Py Socket Server

Enter Your ID and Password

User ID:

Password:

Hello Py Socket Server

Enter Your ID and Password

User ID:

Password:

<http://serverIP:10080>에 접속 시, index.html 과 같은 화면이 나오며 input form 도 정상적으로 출력된다.

- If a user enters ID & password (and push the login button), the server returns "secret.html".
 - You prepare your own "secret.html" file that includes any interesting story (text) about you and related image files (more than 3 images).

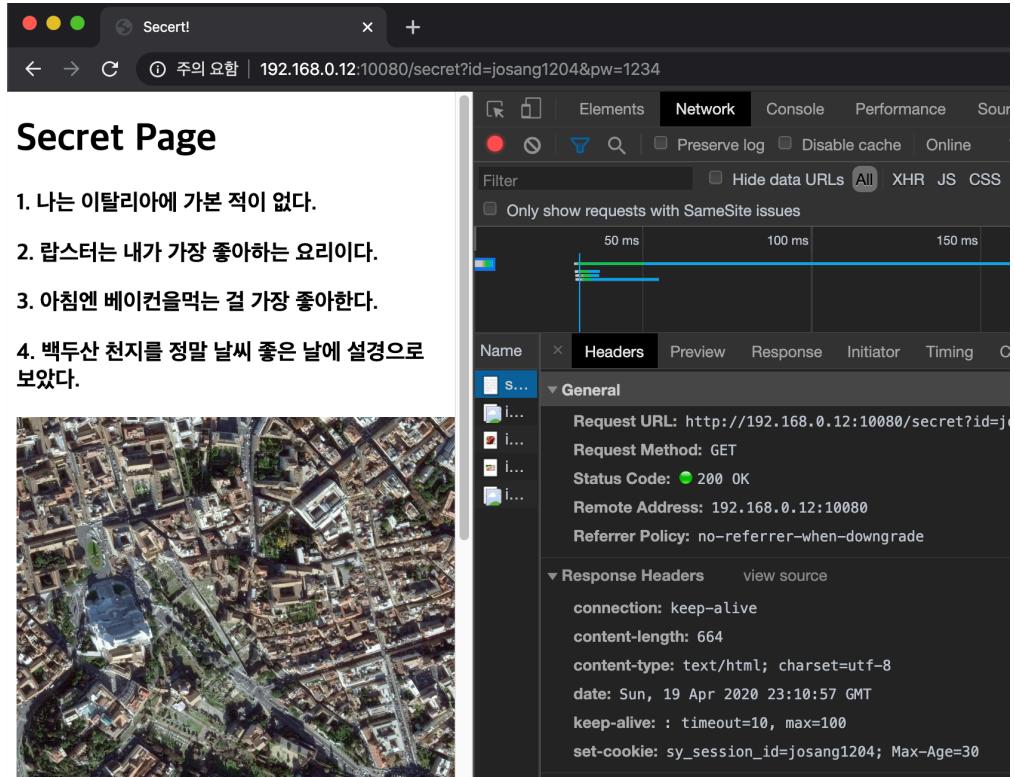


그림 8 ID, PW 정상 입력 후 Secret 페이지 진입

Secret Page

1. 나는 이탈리아에 가본 적이 없다.
2. 랍스터는 내가 가장 좋아하는 요리이다.
3. 아침엔 베이컨을 먹는 걸 가장 좋아한다.
4. 백두산 천지를 정말 날씨 좋은 날에 설경으로 보았다.

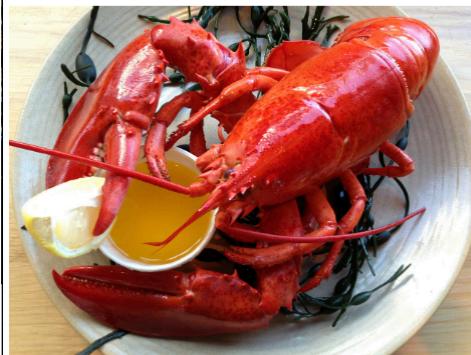


그림 9 그림 4 장 삽입

- Without the login process, if users try to access any (even non-existing) URL directly, the web server program must return a "403 Forbidden" error message.
 - Once you logged in, you can access any content in the web server for 30 seconds. After 30 seconds, your cookie will be expired.
 - After expiration, you have to log in again to access another content in the web server.

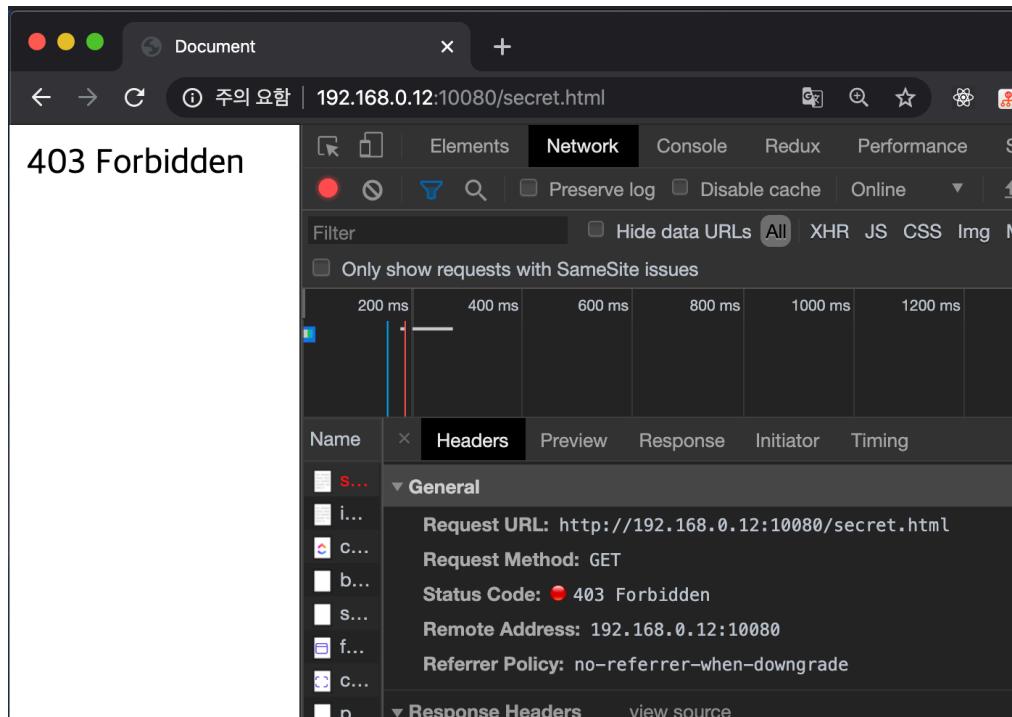


그림 10 로그인 하지 않아 403 forbidden

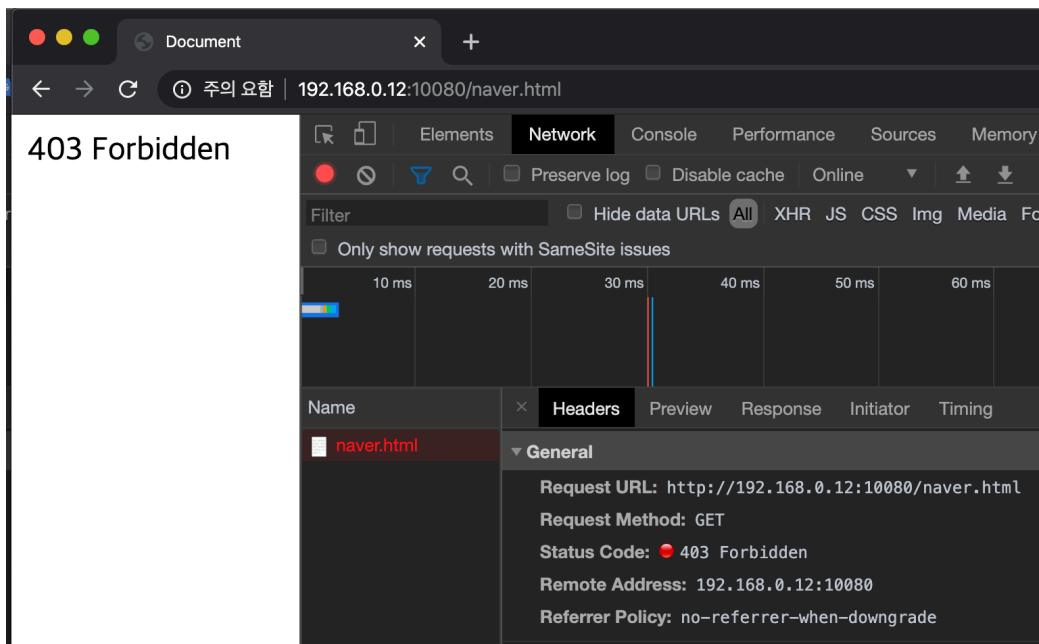


그림 11 로그인 하지 않아 존재하지 않는 경로에도 403 forbidden

- When accessing "cookie.html", you can see "User_ID" in the browser title bar, and display how many seconds left before the cookie expires.

Three browser windows are shown side-by-side, each displaying a "Cookie Page".

- The first window is titled "User: computer". The title bar says "User: computer" and the address bar says "192.168.0.12:10080/cookie.html". The page content says "Hello computer" and "23 seconds left until your cookie expires."
- The second window is titled "User: josang1204". The title bar says "User: josang1204" and the address bar says "192.168.0.12:10080/cookie.html". The page content says "Hello josang1204" and "12 seconds left until your cookie expires."
- The third window is titled "User: firefox". The title bar says "User: firefox" and the address bar says "192.168.0.12:10080/cookie.html". The page content says "Hello firefox" and "15 seconds left until your cookie expires."

A developer tools Network tab is visible on the right side of the bottom window, showing a list of requests. It includes a table with the following data:

Stat...	Met...	Domain
200	GET	192.168..
403	GET	192.168..

그림 12 User ID 별 쿠키 남은 초 표시

4.3 Persistent HTTP mode (10 points)

- Upgrade your web server program to be persistent HTTP mode.

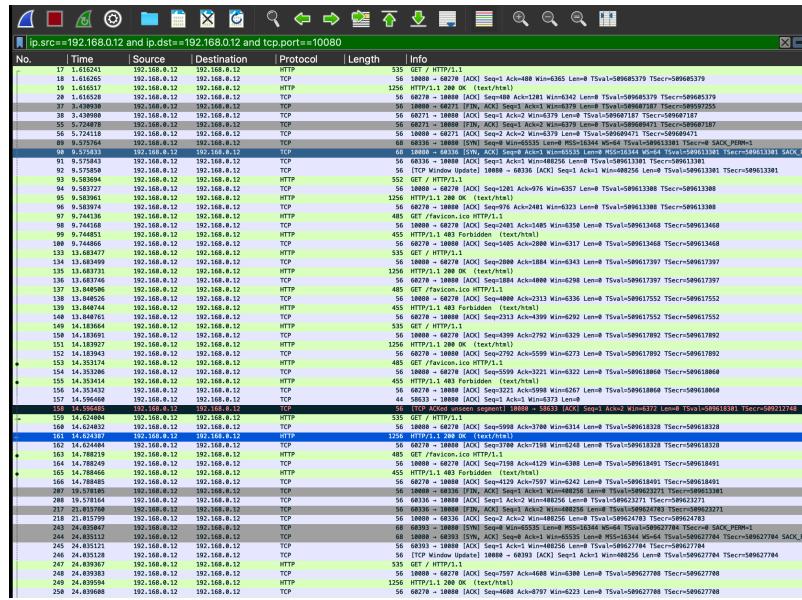


그림 13 Wireshark 팩킷

초기에 handshake 이후 연속적인 요청에 대하여 persistent mode로 connection을 유지하고 있다가 잠시 쉬었다가 다시 요청을 보내니 timeout이 되어 뒤에서 다시 handshake 하는 걸 알 수 있다.

```

TCP payload (1200 bytes)
▼ Hypertext Transfer Protocol
▶ HTTP/1.1 200 OK\r\n
date: Sun, 19 Apr 2020 23:21:40 GMT\r\n
connection: keep-alive\r\n
▶ content-length: 1025\r\n
content-type: text/html; charset=utf-8\r\n
keep-alive:: timeout=10, max=92\r\n
\r\n
[HTTP response 8/11]
[Time since request: 0.000383000 seconds]

```

그림 14 8 번째 요청이라 max 100 이 92로 줄어듦