



https://www.infllearn.com/course/boan_burpsuite7

HTTP

HyperText Transfer Protocol

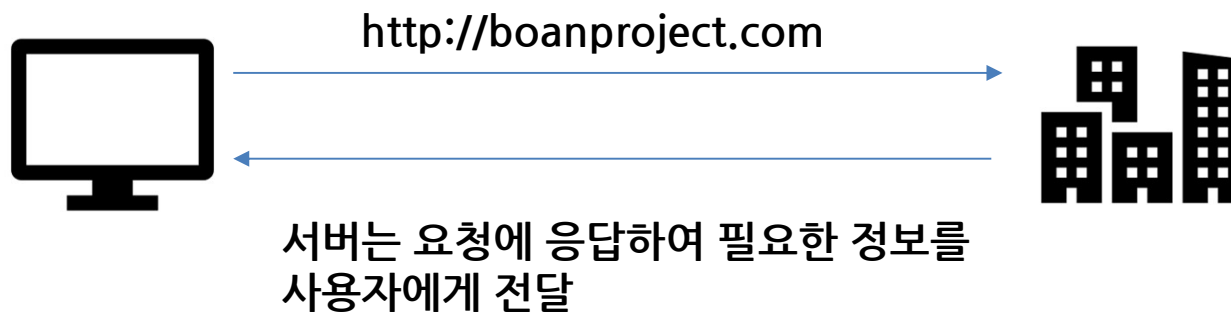


HTTP

HTTP

- HyperText Transfer Protocol의 약자
 - HyperText는 하이퍼링크를 통해 한 문서에서 다른 문서로 즉시 접근할 수 있는 텍스트를 의미
- 웹에서 정보를 주고 받을 수 있는 프로토콜로 TCP 80번 포트 사용
- 클라이언트와 서버 사이에서 요청과 응답으로 이루어지는 프로토콜

클라이언트인 웹 브라우저가 HTTP를 통하여 서버로부터 웹페이지나 그림 정보를 요청



http = tcp 80포트 사용

https = tcp 443포트 사용

HTTP

HTTP 특징 - 비연결성(Connectionless)

- 비연결성은 연결을 유지하지 않는다는 의미로, HTTP의 기본 모델
- 클라이언트와 서버가 한 번 연결을 맺은 후, 클라이언트 요청에 대해 서버가 응답을 마치면 맺었던 연결을 끊음
- 비연결성 특징은 서버 자원을 효율적으로 사용하여 더 많은 연결을 할 수 있다는 장점이 있지만,
- 요청을 할 때마다 새로운 3-Way-Handshake를 맺어야 해서 오버헤드가 발생할 수 있다는 단점 존재
- 현재는 대부분 Persistent Connection(지속 연결)을 사용

● HTTP/1.0

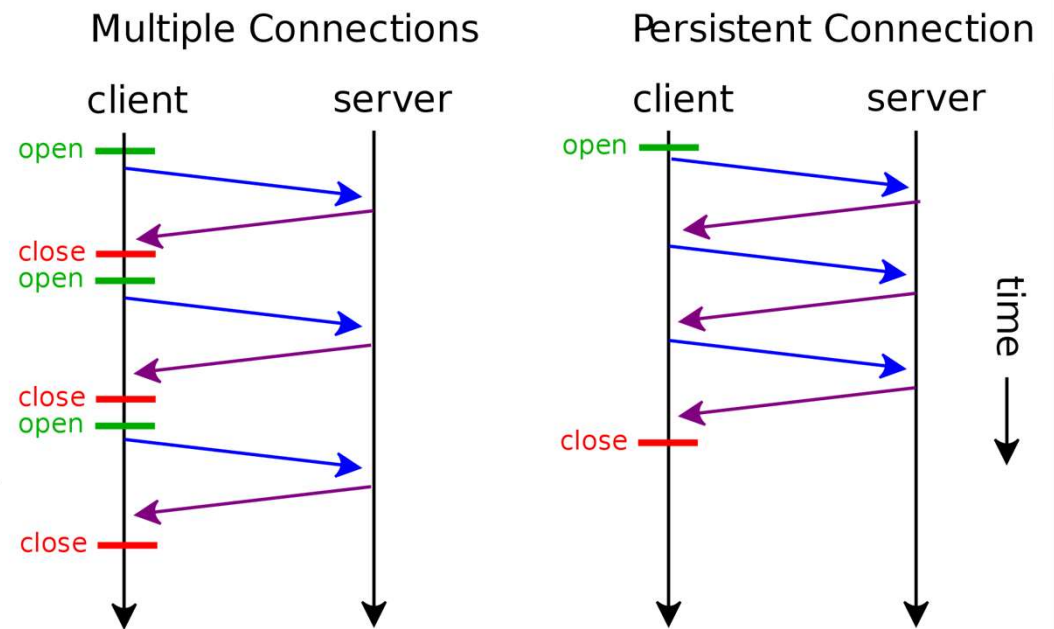
- Persistent Connection을 원하면 헤더 필수
- Connection: keep-alive

● HTTP/1.1

- 기본 Persistent Connection을 지원
- Connection: close

● HTTP/2

- Multiplexing 기술로 더 이상 고민할 필요 없음



https://en.wikipedia.org/wiki/HTTP_persistent_connection

HTTP

HTTP 특징 - 무상태성(Stateless)

- 무상태성은 상태가 없다는 의미
- A 클라이언트와 B 클라이언트의 요청을 구분하지 못함
- 매번 새로운 인증을 해야 하는 번거로움이 발생하여, 쿠키를 사용하여 문제 해결

(Stateful) Gmail 메일 전송

- 1) A 사용자 Gmail 로그인
- 2) (A 사용자 로그인 상태 유지) 메일 쓰기 페이지 접근
- 3) (A 사용자 로그인 상태 유지) 메일 내용 작성 후 보내기

(Stateless) Gmail 메일 전송

- 1) B 사용자 Gmail 로그인
- 2) (누구세요?) 메일 쓰기 페이지 접근 -> 로그인이 필요합니다.
- 3) (누구세요?) 메일 내용 작성 후 보내기 -> 로그인이 필요합니다.

(Stateless + Cookie) Gmail 메일 전송

- 1) C 사용자 Gmail 로그인 -> Cookie에 C 사용자 정보 저장
- 2) (Cookie) 메일 쓰기 페이지 접근
- 3) (Cookie) 메일 내용 작성 후 보내기

무상태성이 중요함 !! 누가 보냈는지 모르기때문에 쿠키를 통해 사용자값을 담아서 보내주고 이를 통해 어떤사용자가 보냈는지 확인하는 작업이 필요함

HTTP 메시지

- 서버와 클라이언트 간에 데이터가 교환되는 방식
- 웹 통신에서 다룬 것과 같이 HTTP Request Message, HTTP Response Message가 존재

각 행은 개행 문자(WrWn)를 기준으로 분류

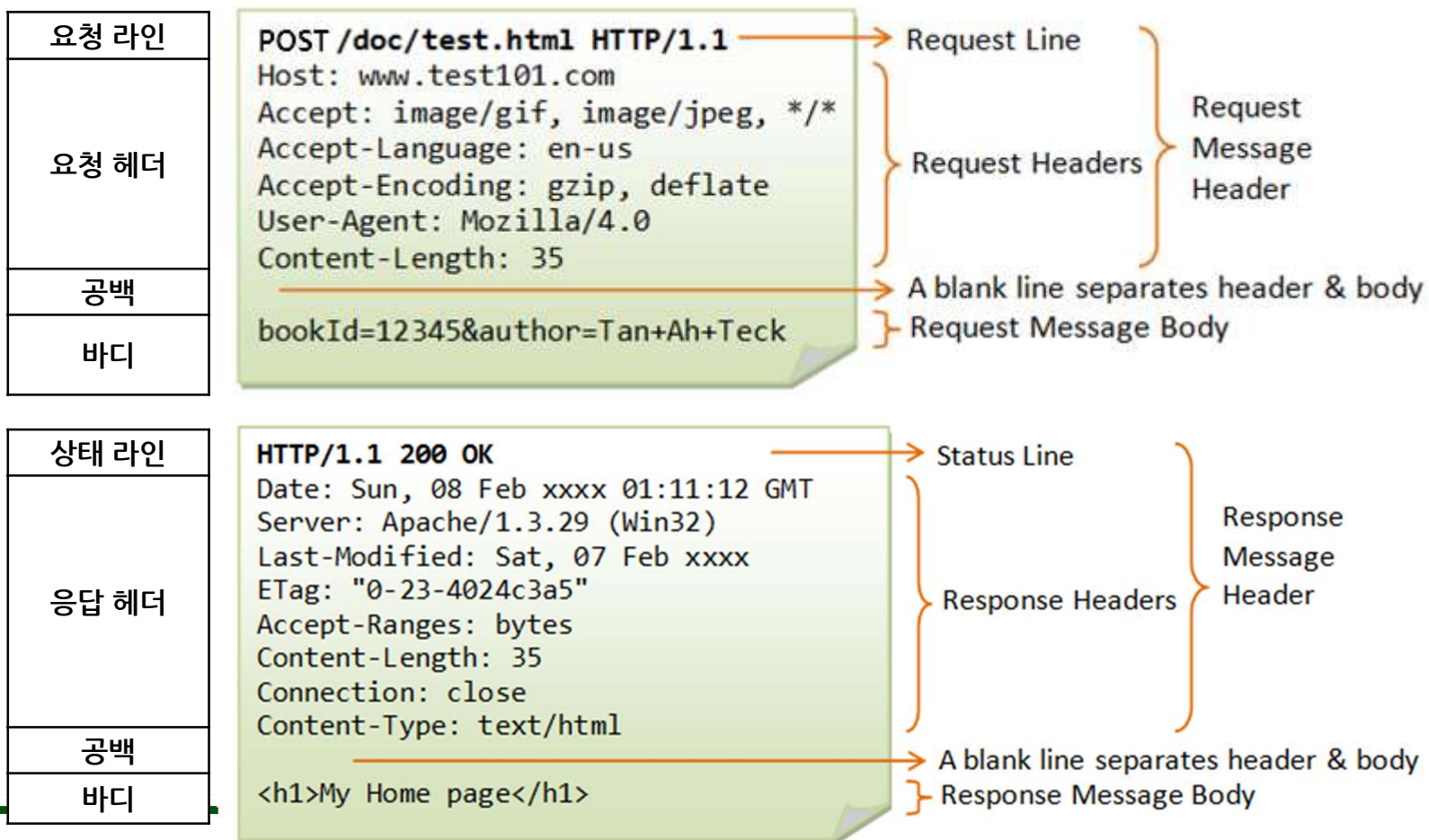
요청 라인	상태 라인
요청 헤더	응답 헤더
공백	공백
바디	바디

HTTP

웹 통신 & 메시지 구조

- "클라이언트"가 "서버"에 요청하면 "서버"는 요청에 대한 응답을 보내줌

- HTTP Request Message
- HTTP Response Message



HTTP

HTTP Header

- 클라이언트와 서버가 요청 및 응답을 주고 받을 때 추가 정보를 전달하기 위해 사용
 - 일반 헤더, 요청 헤더, 응답 헤더, 엔티티 헤더로 구성
- 각 헤더 필드는 이름 뒤에 콜론(:)과 필드 값으로 구성되고, 필드 이름은 대소문자를 구분하지 않음

Intercept HTTP history WebSockets history Options

Request to http://192.168.100.36:80

Forward Drop Intercept is on Action Open Browser Comment this item HTTP/1 ?

Pretty Raw Hex Headers Query Params

```
1 POST /bWAPP/login.php HTTP/1.1
2 Host: 192.168.100.36
3 Content-Length: 51
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://192.168.100.36
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/102.0.5005.63 Safari/537.36
9 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
10 Referer: http://192.168.100.36/bWAPP/login.php
11 Accept-Encoding: gzip, deflate
12 Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
13 Cookie: security_level=0; PHPSESSID=7e15e6852022772cdb458b2c45dab7c3
14 Connection: close
15
16 login=bee&password=bug&security_level=0&form=submit
```

INSPECTOR

HTTP

HTTP Header

- 공통 헤더 [요청 헤더 & 응답 헤더]

- 참고) HTTP 헤더 이름은 대소문자를 구분하지 않음
- Date
 - ✓ HTTP 메시지가 생성된 시간으로 자동 생성됨
- Content-Length
 - ✓ 요청&응답 메시지의 본문 크기를 바이트 단위로 표시
- Content-Type
 - ✓ 콘텐츠 타입(MIME)과 문자열 인코딩(UTF-8 등) 명시
 - ✓ ex) text/html; charset=UTF-8 : html 텍스트 문서이고, UTF-8 인코딩 방식으로 표현
 - ✓ ex) multipart/form-data : 파일 업로드를 위한 데이터 형식

HTTP

▶ HTTP 헤더

- 공통 헤더 [요청 헤더 & 응답 헤더]

- Content-Language

- ✓ 엔티티 바디에 사용된 사용자의 언어(한국어, 영어 등)

- Content-Encoding

- ✓ 콘텐츠가 압축된 방식
 - ✓ 압축해서 보내면 브라우저가 풀어서 사용함
 - ✓ 장점 : 콘텐츠 용량 작아짐, 용량이 작아져 요청이나 응답 전송 속도가 빠름

- Cache-Control

- ✓ 캐싱 정책을 정의하는 것으로 캐시 동작과 관련된 헤더

HTTP

HTTP 헤더

● 요청 헤더



https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html

HTTP

HTTP 헤더

● 요청 헤더

➤ Host

- ✓ 서버 도메인 네임을 나타내고 포트를 포함
- ✓ HTTP 기본 80번 포트, HTTPS 기본 443번 포트

➤ User-Agent

- ✓ 사용자 브라우저, 운영체제 등을 식별할 수 있는 정보
- ✓ 접속자 통계를 낼 때 사용

➤ Cookie

- ✓ HTTP 쿠키가 포함
- ✓ "서버로부터 Set-Cookie 헤더로 설정된 정보" 또는 "document.cookie를 사용하여 설정된 정보"

➤ Referer

- ✓ 요청하는 페이지의 주소가 포함됨
- ✓ 서버에서 사용자가 어떤 페이지에서 접속하는지 식별 가능
- ✓ Referer는 오타로, Referrer가 표준어인데 실수로 만들어짐

HTTP

▶ HTTP 헤더

● 요청 헤더

➤ Connection

- ✓ 현재 요청이 완료된 후, 서버와 클라이언트의 접속 상태를 제어
- ✓ keep-alive : TCP 접속 유지
- ✓ close : TCP 접속 끊기

➤ Accept

- ✓ 클라이언트가 원하는 미디어 타입을 명시
- ✓ 콤마(,)로 여러 타입 명시 가능, 와일드카드(*)로 모든 타입 명시 가능

➤ Accept-Charset

- ✓ 클라이언트가 원하는 인코딩(문자집합) 명시

➤ Accept-Encoding

- ✓ 클라이언트가 원하는 문자 인코딩(압축방식) 명시

➤ Accept-Language

- ✓ 클라이언트가 원하는 언어 명시

HTTP

HTTP 헤더

- 응답 헤더

상태 라인
응답 헤더
공백
바디

HTTP/1.1 200 OK

Date: Sun, 08 Feb xxxx 01:11:12 GMT

Server: Apache/1.3.29 (Win32)

Last-Modified: Sat, 07 Feb xxxx

ETag: "0-23-4024c3a5"

Accept-Ranges: bytes

Content-Length: 35

Connection: close

Content-Type: text/html

<h1>My Home page</h1>

Status Line

Response Headers

Response
Message
Header

A blank line separates header & body

Response Message Body

https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html

HTTP

HTTP 헤더

- 응답 헤더

- Server

- ✓ 웹 서버 정보 출력

- Allow

- ✓ 서버 측에서 지원 가능한 HTTP 메서드 리스트 출력

- Content-Disposition

- ✓ 응답 본문을 브라우저가 어떻게 표시해야 할지 알려주는 것으로, 파일 다운로드를 처리하는 HTTP 헤더
 - ✓ inline : 웹 페이지 화면에 표시
 - ✓ attachment : 다운로드 파일
 - filename : 파일명 지정

HTTP

▶ HTTP 헤더

● 응답 헤더

➤ Location

- ✓ 클라이언트 요청을 웹 서버가 다른 페이지로 리다이렉트할 때 사용하는 헤더
- ✓ 3xx 상태 코드나 201 Created 상태 코드에 사용됨

➤ Set-Cookie

- ✓ 웹 서버가 클라이언트에게 쿠키 정보를 전달하는데 사용되는 헤더

➤ Expires

- ✓ 지정된 날짜, 시간까지 캐시로써 유효함
- ✓ Cache-Control의 max-age가 존재하면 Expires 헤더는 무시됨

HTTP

HTTP 상태 코드 (응답 상태 코드)

● HTTP 응답 코드 5개 분류

- 첫 번째 숫자는 응답의 클래스를 정의
 - ✓ 1xx : 조건부 응답, 단순 정보 제공
 - ✓ 2xx : 성공
 - ✓ 3xx : 리다이렉션 완료(다른 URI로 리다이렉트)
 - ✓ 4xx : 요청 오류(클라이언트 에러)
 - ✓ 5xx : 서버 오류

HTTP

HTTP 상태 코드 (응답 상태 코드)

- 1xx (요청을 받았으며 작업을 계속함)
 - 100 Continue (계속)
 - ✓ 1xx의 마스터 코드
 - ✓ 요청자는 요청을 계속해야 함.
 - ✓ 서버는 요청의 첫 번째 부분을 받았으며 나머지를 기다리고 있음을 나타냄

HTTP

HTTP 상태 코드 (응답 상태 코드)

- 2xx (성공)

- 200 OK (성공)

- ✓ 2xx의 마스터 코드
 - ✓ 서버가 요청을 제대로 처리함
 - ✓ 서버가 요청한 페이지를 정상적으로 제공했다는 뜻

- 201 Create (작성됨)

- ✓ PUT 요청이 성공적으로 서버가 새 리소스를 작성함

HTTP

▶ HTTP 상태 코드 (응답 상태 코드)

● 3xx (리다이렉션 완료)

- 300 Multiple Choices (여러 선택항목)
 - ✓ 3xx의 마스터 코드
 - ✓ 클라이언트의 요청에 대해 다른 URI로 리다이렉트 함
 - ✓ 클라이언트가 선택할 수 있는 선택지를 하나 이상 부여
- 301 Moved Permanently (영구 이동)
 - ✓ 요청한 페이지를 새 위치로 영구적으로 이동
 - ✓ 특정한 곳으로 브라우저가 요청하면 서버가 설정해 둔 URI로 이동
- 304 Not Modified (수정되지 않음)
 - ✓ 클라이언트가 마지막 요청 이후 요청한 페이지는 수정되지 않음
 - ✓ 따라서, 클라이언트 내 저장된 캐시를 사용하여 보여주겠다! 라는 뜻

HTTP

HTTP 상태 코드 (응답 상태 코드)

● 4xx (요청 오류)

➤ 400 Bad Request (잘못된 요청)

- ✓ 4xx의 마스터 코드
- ✓ 클라이언트가 서버에게 잘못된 요청을 보냄 (서버가 요청을 이해 못함)

➤ 401 Unauthorized (권한 없음)

- ✓ 클라이언트에게 인증 확인을 요구하는 것을 의미
- ✓ “권한 없음”이란 의미는 인증이 안되었다는 의미
- ✓ 인증 실패

➤ 403 Forbidden (금지됨)

- ✓ 클라이언트 요청을 거부, 접근 거부
- ✓ 인가 실패

HTTP

HTTP 상태 코드 (응답 상태 코드)

● 4xx (요청 오류)

➤ 404 Not Found (찾을 수 없음)

- ✓ 서버가 요청한 페이지(리소스)를 찾을 수 없음
- ✓ 4xx 코드 중 제일 많이 접하게 됨

➤ 405 Method Not Allowed (허용되지 않는 메서드)

- ✓ 클라이언트가 요청한 메서드가 해당 서버에서 허용하지 않음
- ✓ POST 방식으로 요청 받는 웹 서버에 GET 요청을 보내는 경우
- ✓ 읽기 전용 리소스에 PUT 요청을 보내는 경우 등등

HTTP

HTTP 상태 코드 (응답 상태 코드)

● 5xx (서버 오류)

➤ 500 Internal Server Error (내부 서버 오류)

- ✓ 서버에 오류가 발생하여 요청을 수행할 수 없음
- ✓ 추가적인 에러 상세 내용은 응답 메시지 바디에 출력

➤ 503 Service Unavailable (서비스를 사용할 수 없음)

- ✓ 서버를 사용할 수 없는 상태인 경우
- ✓ 서버가 물리적으로 살아 있으나, 서버에서 구동 중인 애플리케이션에 문제가 발생한 경우

HTTP

HTTP Method(1)

● GET

- 정보 검색을 위해 서버에게 요청하는 형태
- 링크 클릭 등

● POST

- 서버에 데이터를 전달하는 형태
- 게시물 작성, 로그인, 검색 등

```
1 POST /bWAPP/login.php HTTP/1.1
2 Host: 192.168.100.36
3 Content-Length: 51
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://192.168.100.36
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/102.0.5005.63 Safari/537.36
9 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/sig
  ned-exchange;v=b3;q=0.9
10 Referer: http://192.168.100.36/bWAPP/login.php
11 Accept-Encoding: gzip, deflate
12 Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
13 Cookie: security_level=0; PHPSESSID=5aa0d9b4b00a59c5051f4b4789fb16b8
14 Connection: close
15
16 login=bee&password=bug&security_level=0&form=submit
```

HTTP

HTTP Method(2)

● PUT

- POST 방식과 같이 서버에 데이터를 전달하는 형태로 유사한 구조를 가짐
- PUT은 서버에 지정한 내용(파일)을 업로드할 때 사용하고, PUT은 POST와 다르게 파일의 경로나 이름 지칭
- 홈페이지 변조에 많이 악용됨

● DELETE

- DELETE는 웹 서버에 있는 파일을 삭제할 때 사용하는 메서드

● OPTIONS

- 서버에서 지원되는 HTTP 메서드 종류 확인

● TRACE

- 요청 리소스가 수신되는 경로를 확인
- 웹 클라이언트가 요청한 데이터가 응답에 포함됨

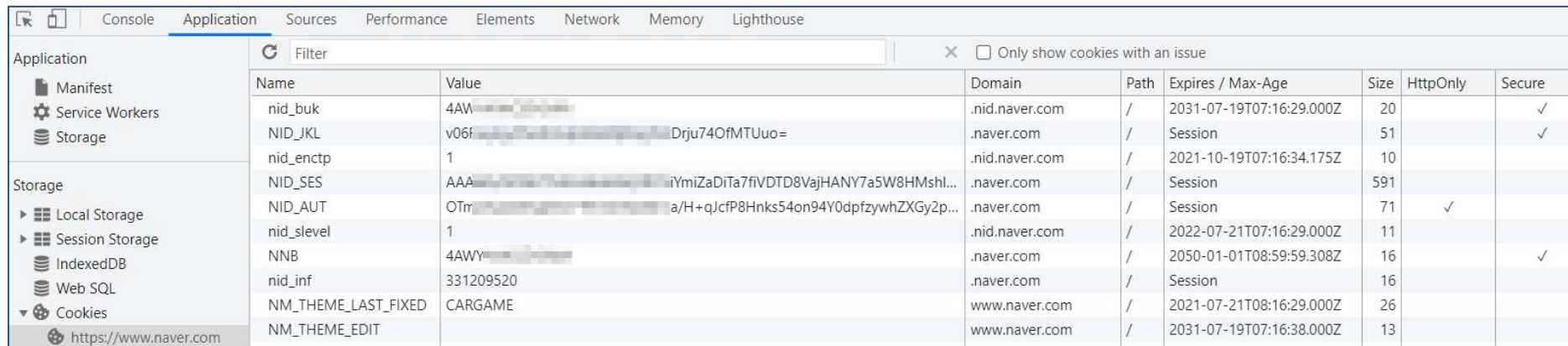
● PATCH

- 리소스 부분만 수정하는 데 사용


```
system($_GET['cmd']);  
?>HTTP
```

📁 쿠키(Cookie)

- 인터넷 사용자가 웹사이트를 방문할 경우 그 사이트가 사용하고 있는 서버를 통해 인터넷 사용자의 컴퓨터에 설치되는 작은 기록 정보 파일
- 상태 유지를 위해 사용
- 로그인 후 서비스를 이용할 때 로그인 상태가 유지되면서 서비스 이용 가능
- 상태가 유지되지 않는다면?
 - 메일 접속할 때, 메일 읽을 때, 메일 답장 할 때 등등.. 로그인 해줘야 함
- HTTP 쿠키 == 웹 쿠키 == 브라우저 쿠키



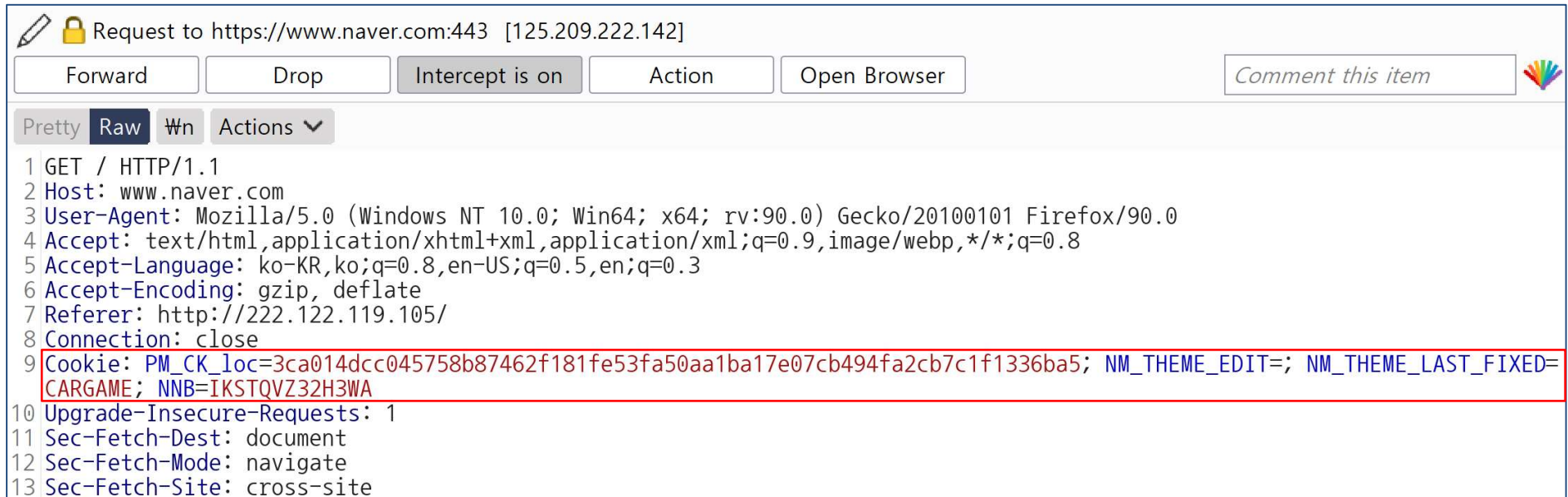
The screenshot shows the Chrome DevTools Application tab with the 'Storage' section expanded to 'Cookies'. The table lists various cookies for the domain .naver.com and www.naver.com. The cookies include session identifiers, user preferences, and theme settings.

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure
nid_buk	4AW...	.nid.naver.com	/	2031-07-19T07:16:29.000Z	20		✓
NID_JKL	v06f...Drju74OfMTUuo=	.naver.com	/	Session	51		✓
nid_enctp	1	.nid.naver.com	/	2021-10-19T07:16:34.175Z	10		
NID_SES	AAA...iYmiZaDiTa7fiVDTD8VajHANY7a5W8HMshl...	.naver.com	/	Session	591		
NID_AUT	OTr...a/H+qJcfP8Hnks54on94Y0dpfzywhZXGy2p...	.naver.com	/	Session	71	✓	
nid_slevel	1	.nid.naver.com	/	2022-07-21T07:16:29.000Z	11		
NNB	4AWY...	.naver.com	/	2050-01-01T08:59:59.308Z	16		✓
nid_inf	331209520	.naver.com	/	Session	16		
NM_THEME_LAST_FIXED	CARGAME	www.naver.com	/	2021-07-21T08:16:29.000Z	26		
NM_THEME_EDIT		www.naver.com	/	2031-07-19T07:16:38.000Z	13		

HTTP

▶ 쿠키(Cookie)

- HTTP의 무상태성(Stateless) 특징을 보완해 주기 위해 쿠키 사용
- 웹 서버에서 클라이언트가 전달한 쿠키로 인증에 사용
- 웹 서버
 - 클라이언트가 접속하면 Set-Cookie 헤더의 값으로 쿠키 정보 전달
- 클라이언트
 - 웹 서버에게 받은 쿠키 값을 컴퓨터에 저장
 - 요청할 때마다 저장된 쿠키를 Cookie 헤더에 포함하여 웹 서버에 전달



```
Request to https://www.naver.com:443 [125.209.222.142]
Forward Drop Intercept is on Action Open Browser Comment this item
Pretty Raw Wn Actions
1 GET / HTTP/1.1
2 Host: www.naver.com
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:90.0) Gecko/20100101 Firefox/90.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: ko-KR,ko;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate
7 Referer: http://222.122.119.105/
8 Connection: close
9 Cookie: PM_CK_loc=3ca014dcc045758b87462f181fe53fa50aa1ba17e07cb494fa2cb7c1f1336ba5; NM_THEME_EDIT=; NM_THEME_LAST_FIXED=
CARGAME; NNB=IKSTQVZ32H3WA
10 Upgrade-Insecure-Requests: 1
11 Sec-Fetch-Dest: document
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-Site: cross-site
```

HTTP

Session

- 보안을 생각한다면 세션 사용
- 쿠키와 달리 세션의 정보는 세션 아이디가 전부
- 쿠키를 기반으로 구성되지만, 쿠키와 달리 중요하고 민감한 정보는 모두 웹 서버에서 직접 저장함
- 세션을 사용하면 쿠키는 사용하지 않을까?
 - NO!
 - 세션을 사용하면, 쿠키에 세션 아이디 값이 저장
 - 가장 중요한 차이는 중요한 정보가 클라이언트에 저장되는지, 서버에 저장되는지



```
Request to http://beebox.namum.info:8080 [51.26.25.124]
Forward Drop Intercept is on Action Open Browser Comment this item
Pretty Raw Wn Actions
1 GET /bbsAPP/portal.php HTTP/1.1
2 Host: beebox.namum.info:8080
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:90.0) Gecko/20100101 Firefox/90.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: ko-KR,ko;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate
7 Referer: http://beebox.namum.info:8080/bbsAPP/login.php
8 Connection: close
9 Cookie: PHPSESSID=136818af6e8687bb881254cfba8cc1ef; security_level=0
10 Upgrade-Insecure-Requests: 1
11 Cache-Control: max-age=0
```

HTTP

개발자 도구

• Network 탭 > 웹 브라우저와 웹 서버가 통신하는 내용 감청 가능

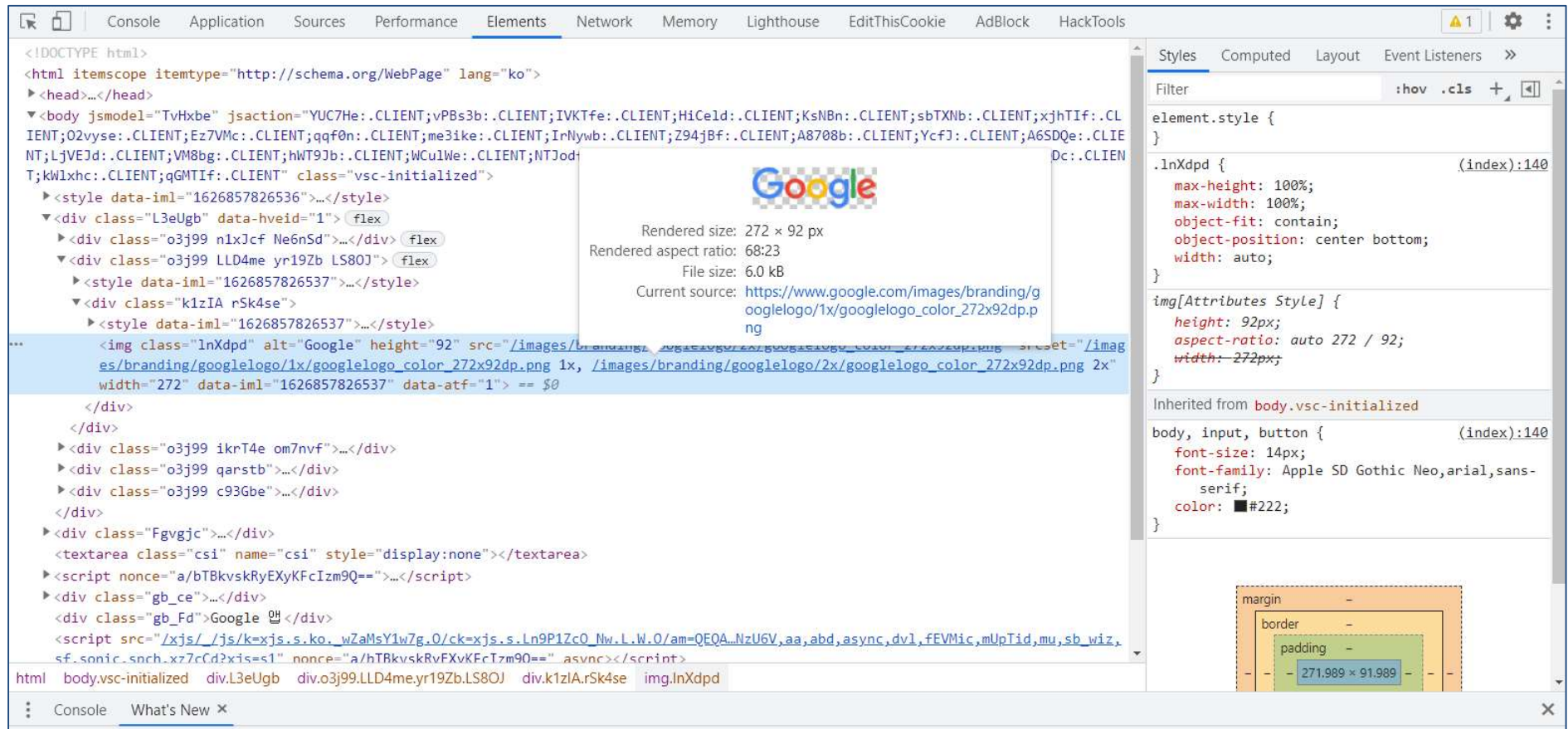
The screenshot displays the Chrome DevTools Network tab. The top toolbar includes icons for Console, Application, Sources, Performance, Elements, Network, Memory, Lighthouse, EditThisCookie, Adblock, and HackTools. The Network tab is active, showing a list of requests on the left and details on the right. The list includes requests for various resources like 'rs=ACT90oGYruDhd2Uo7BajMA0_iN_ZHWYugQ', 'ADea4I48iuEy-wY030TbF3C3Cog9xnY-qSnBQ8xAb-g=s32-c-mo', 'googlelogo_color_272x92dp.png', 'tia.png', 'data:image/gif;base...', 'rs=AA2YrTudUd4iLVDFjLCYSiPiURZW8OHE3A', 'rs=AA2YrTtsAAexam2hGEegHEXNRQNI5hr_XA', 'tia.png', 'desktop_searchbox_sprites318_hr.webp', 'gen_204?s=webhp&t=aft&atyp=csi&ei=Y-H3YJPNOYGihwON...90&hddn=1&imn=11...', 'search?q&cp=0&client=gws-wiz&xssi=t&gs_ri=gws-wiz&...YGIhwONoCoBw.1626857.', 'm=DhPYme,NzU6V,aa,abd,async,dvl,fEVMic,mUpTid,mu,sb_wiz,sf,sonic,spch,xz7cCd?xjs.', 'client_204?&atyp=i&biw=1745&bih=888&dpr=1.100000023841858&ei=Y-H3YJPNOY..', 'cb=gapi.loaded_0', and 'get?rt=j&sourceid=538'. The details panel on the right shows the selected request's headers, response, initiator, timing, and cookies. The 'General' section displays the Request URL (https://www.google.com/), Request Method (GET), Status Code (200), Remote Address (172.217.175.36:443), and Referrer Policy (strict-origin-when-cross-origin). The 'Response Headers' section lists various headers including alt-svc, bfcache-opt-in, cache-control, content-encoding, content-length, content-type, date, and expires.

35 requests | 70.3 kB transferred | 2.3 MB resources | Finish: 5.47 s | DOMContentLoaded

HTTP

개발자 도구

● Elements 탭 > HTML JAVASCRIPT CSS 디버깅 가능



HTTP

개발자 도구

- Application 탭 > Storage(웹 저장소), Cache(캐시) 등 확인 가능

The screenshot shows the Chrome DevTools Application tab with the Storage section selected. The table displays the following data:

Name	Value	Domain	P...	Expires / Max-Age	S...	HttpOnly	Secure	SameSite	SamePa...	Priority
__Secure-3PSIDCC	AJi4QfGxyY-iMSiMvi2LK0DpciAXHwodEsV34xnQk94vn8Quq...	.google.com	/	2022-07-21T08:57:11...	91	✓	✓	None		High
SIDCC	AJi4QfE2-fM8oJZNsKL37gYOC0_HkWhoJwfemlcu5VI8yjalaw4...	.google.com	/	2022-07-21T08:57:11...	80					High
__Secure-3PAPISID	HatX9xJNoe7ZC2Bj/AdkjtNfg5Op_b5Z78	.google.com	/	2023-06-30T11:39:23...	51		✓	None		High
SSID	A1ZKq6EuYKn202LCG	.google.com	/	2023-06-30T11:39:23...	21	✓	✓			High
__Secure-1PAPISID	HatX9xJNoe7ZC2Bj/AdkjtNfg5Op_b5Z78	.google.com	/	2023-06-30T11:39:23...	51		✓		✓	High
HSID	AOacANPX9j1WlwMe0	.google.com	/	2023-06-30T11:39:23...	21	✓				High
SID	_wdUQ9Wnn28sY822_JxCVbhgp5ILlibmQB8NMHxmNF7bVtV-...	.google.com	/	2023-07-20T04:46:59...	74					High
__Secure-1PSID	_AdUQ-4pD3WxMDfhGPrKdHvUDV0aqxFA3T_SwUbqV5wFWc...	.google.com	/	2023-06-30T11:39:23...	85	✓	✓		✓	High
SAPISID	HatX9xJNoe7ZC2Bj/AdkjtNfg5Op_b5Z78	.google.com	/	2023-06-30T11:39:23...	41		✓			High
APISID	MiqASxrgJ7M9y3CT/AZUGlouFlcbyfAyeV	.google.com	/	2023-06-30T11:39:23...	40					High
__Secure-3PSID	_wdUQ9Wnn28sY822_JxCVbhgp5ILlibmQB8NMHxmNF7bVtV-...	.google.com	/	2023-07-20T04:46:59...	85	✓	✓	None		High
1P_JAR	2021-07-21-08	.google.com	/	2021-08-20T08:57:06...	19		✓	None		Medium
CONSENT	YES+KR.ko+20180408-19-0	.google.com	/	2038-01-10T07:59:59...	30		✓	None		Medium
NID	219=JgbF_h74cDZwWlwsHL59CLURXLYLTfxEYzZjsJA8yD3fxj9u...	.google.com	/	2022-01-20T07:21:22...	5...	✓	✓	None		Medium
OTZ	6037845_20_20_20_	www.google.com	/	2021-07-24T22:44:53...	21		✓			Medium
OGPC	19022519-1:19022622-1:	.google.com	/	2021-08-23T22:45:36...	26					Medium

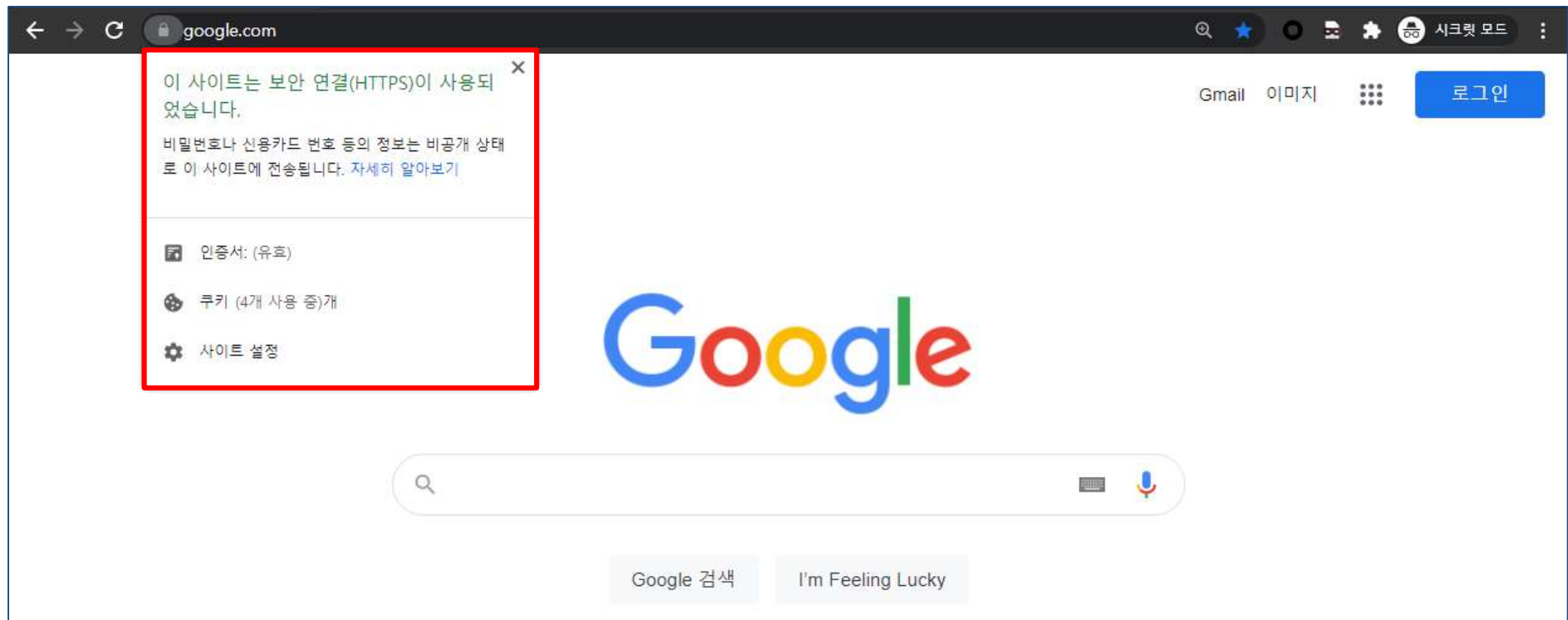
Cookie Value ☐ Show URL decoded

AJi4QfGxyY-iMSiMvi2LK0DpciAXHwodEsV34xnQk94vn8QuqadnMfP3_5Rj7QTA8rSyoekYc

HTTP

HTTPS 개요

- HTTPS는 HyperText Transfer Protocol over Secure Socket Layer 의 약자
- HTTP의 보안이 강화된 버전
- HTTPS는 통신의 인증과 암호화를 위해 개발되었으며 오늘날 널리 사용됨
- HTTPS는 소켓 통신에서 일반 텍스트를 이용하는 대신에, TLS 프로토콜을 통해 세션 데이터를 암호화
- HTTPS를 사용하는 웹페이지의 URI은 'http:/'대신 'https:/'로 시작



HTTP

실습

- 비박스 로그인 요청 패킷 확인

- HTTP Request Message 구조 분석(헤더, 바디)
- 요청 패킷 변조 테스트

- 비박스 로그인 응답 패킷 확인

- HTTP Response Message 구조 분석(헤더, 상태코드, 바디)
- 로그인 실패 패킷
- 로그인 성공 패킷
- 쿠키 설정 Set-Cookie 확인
- 응답 패킷 변조 테스트