HTTP & HTTPS

HTTP?

HTTP는 클라이언트와 서버가 서로 어떻게 통신할지를 표준화하는 **TCP/IP 기반의 응용 계층 통신 프로토콜** 인터넷을 통해 어떻게 내용이 요청되고 전송되는지 정의합니다. 응용 계층 프로토콜이란, 단순히 호스트(클라이언트와 서버) 간의 통신 방식을 표준화하는 추상 계층을 의미합니다. HTTP 자체는 클라이언트와 서버 간의 요청과 응답을 TCP/IP에 의존합니다. 기본적으로 TCP 포트 80이 사용되지만, 다른 포트도 사용할 수 있습니다. 그러나, HTTPS는 포트 443을 사용합니다.

HTTP 주요 특징

1. **Stateless:** 서버는 이전의 요청 상태를 기억하지 않는다. 필요한 정보는 클라이언트에서 계속 보내줘야 한다.
2. **클라이언트 서버 구조:** 클라이언트와 서버가 나눠지지 않은 구조에서 Http 프로토콜을 통해 `클라이언트 ↔ 서버`를 Request, Response를 보내는 구조가 됐다. 이러한 분할을 통해 각각의 역할에 더욱더 적합한 기능에 집중해 개발할 수 있게 됐다.

HTTP 기본 매커니즘

클라이언트가 `요청(Request)`하면 서버가 `응답(Response)`하는 것 클라이언트가 서비스 포트에 HTTP 요청을 전송하면 이를 해석해 적절한 응답을 반환

HTTP Request, Response 개념 정리

Request

클라이언트가 서버에게 원하는 걸 요청하는 단계

- **Request Line:** 어떤 메서드로 어디로 보낼지를 알려준다 ex) `GET /products HTTP/1.1`
- **Header:** 요청 시 필요할 수 있는 옵션(토큰, 쿠키 등..)
- **Body:** 요청 시 필요한 데이터가 있다면 사용

Response

서버가 요청을 처리한 후 결과값에 대해 반환

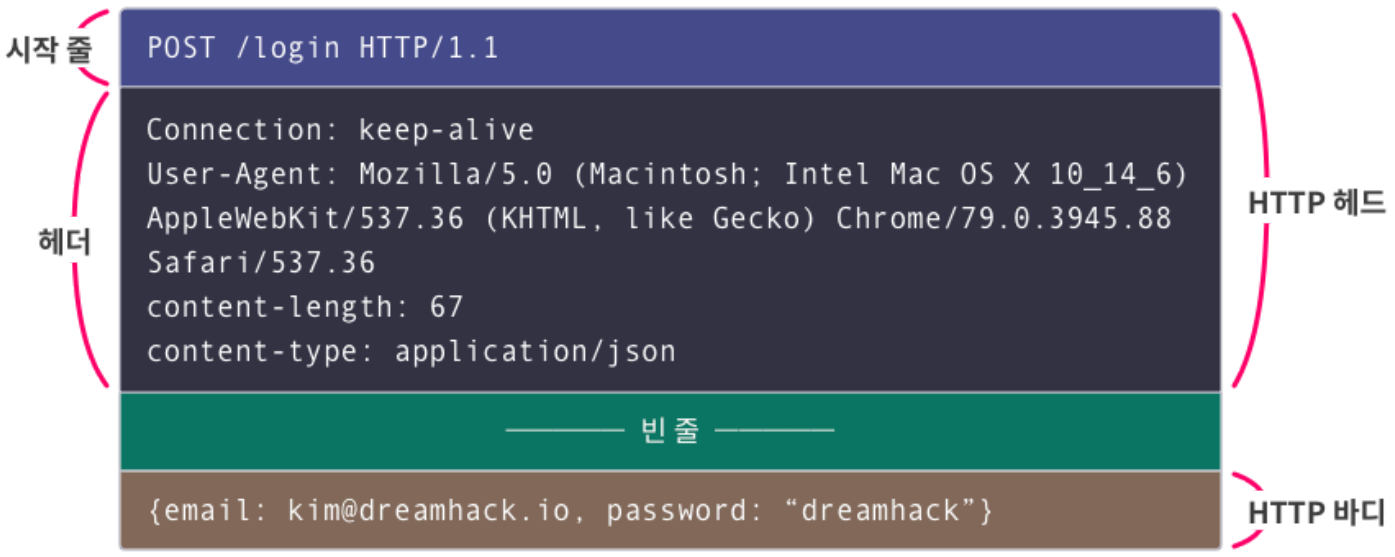
- **Status Line:** 주문의 상태 값에 대한 것을 반환 (200 OK, 201 CREATED, 404 Not Found)
- **Header:** 결과에 붙는 옵션(응답 타입, 길이 ...)
- **Body:** 실제 결과에 대한 데이터(JSON, HTML, File...)

HTTP 메시지

`HTTP Request`, `HTTP Response`으로 분류된다. 이들은 `HTTP 헤더`와 `바디`로 구성된다는 공통점을 가진다

HTTP 헤더

- 첫 줄은 시작 줄, 나머지 줄은 헤더라고 부른다. 헤더의 끝은 빈 줄로 나타낸다



HTTPS 기본 개념 설명

HTTPS(Hypertext Transfer Protocol Secure)는 클라이언트와 서버 간의 데이터 전송을 안전하게 하기 위해 설계된 HTTP의 확장 버전입니다. **SSL/TLS 프로토콜을 통한 암호화**를 사용하여 데이터의 기밀성, 무결성 및 신뢰성을 보장합니다. 이를 통해 로그인 정보나 결제 정보와 같은 민감한 정보가 공격자에 의해 가로채거나 변조되는 것을 방지합니다. HTTPS는 웹 애플리케이션의 보안에 필수적이며, 특히 사용자 데이터를 다루는 대부분의 웹사이트에서 표준이 되었습니다. 이는 중간자 공격과 도청을 방지하는 데 도움이 됩니다.

네트워크 포트와 서비스 포트

네트워크 포트

- 네트워크에서 서버와 클라이언트가 정보를 교환하는 추상화된 장소를 의미한다. 포트는 항구라는 의미가 있으며 클라이언트가 서버의 포트에 데이터를 내려놓고, 서버가 클라이언트에 보낼 데이터를 실어서 돌려보내는 장면을 연상하면 된다.

서비스 포트

- 서비스 포트는 네트워크 포트 중에서 특정 서비스가 점유하고 있는 포트를 의미한다.

포트 데이터 교환 방식

- 포트로 데이터를 교환하는 방식은 전송 계층의 프로토콜을 따른다. 대표적으로 `TCP`와 `UDP`가 존재하며 `TCP`로 데이터를 전송하려는 서비스에 `UDP` 클라이언트가 접근하면 데이터 교환이 불가(반대의 경우도 마찬가지)
- 윈도우나 리눅스, 맥 운영체제는 0번부터 65535번까지의 네트워크 포트를 사용, 0 ~ 1023번 포트는 잘 알려진 포트 또는 `특권 포트`라고 한다. 대표적으로 22번 포트의 `SSH`, 80 `HTTP`, 443 `HTTPS`가 할당돼 있다.

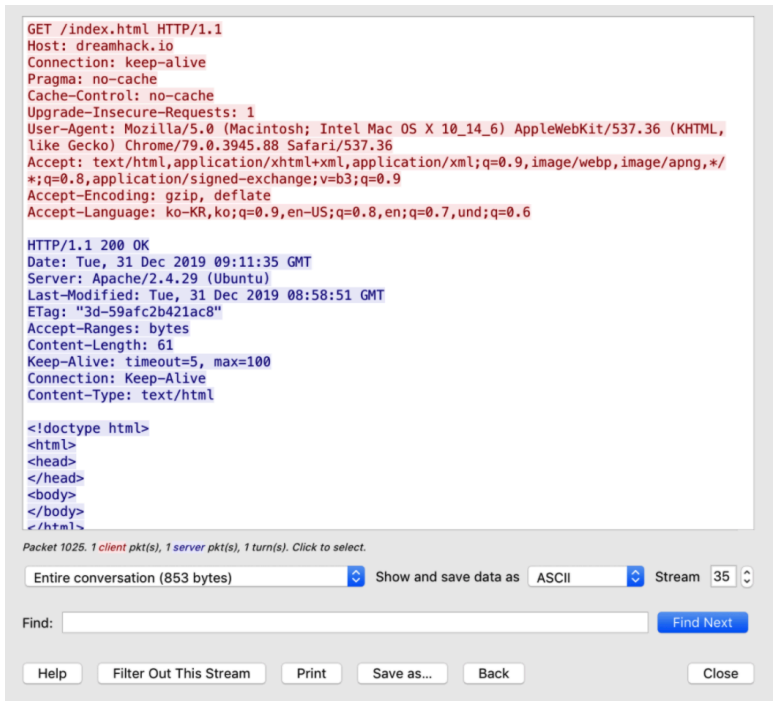
그래서 HTTPS는?

기본적인 HTTP의 응답과 요청은 평문으로 전달된다. 로그인 할 때 전송한 POST 요청을 중간에 탈취하면 이용자의 계정이 탈취 당할 위험이 있다.

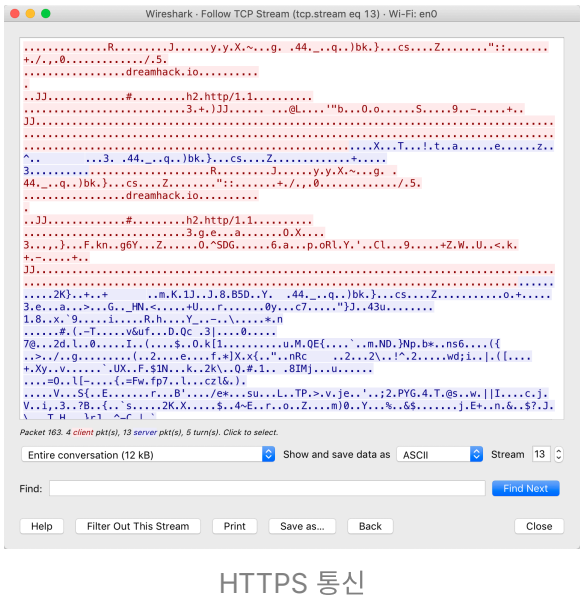
TLS(Transport Layer Security)

HTTPS를 사용 시 TLS를 통해 서버와 클라이언트 사이에 오가는 모든 HTTP 메시지를 암호화한다. 중간에 메시지를 탈취해도 이를 해석하는 것은 불가능하다.

HTTPS가 적용된 메시지의 모습



HTTP 통신



HTTPS 통신