02. Http / Https

컴퓨터공학부 201911228 홍지우

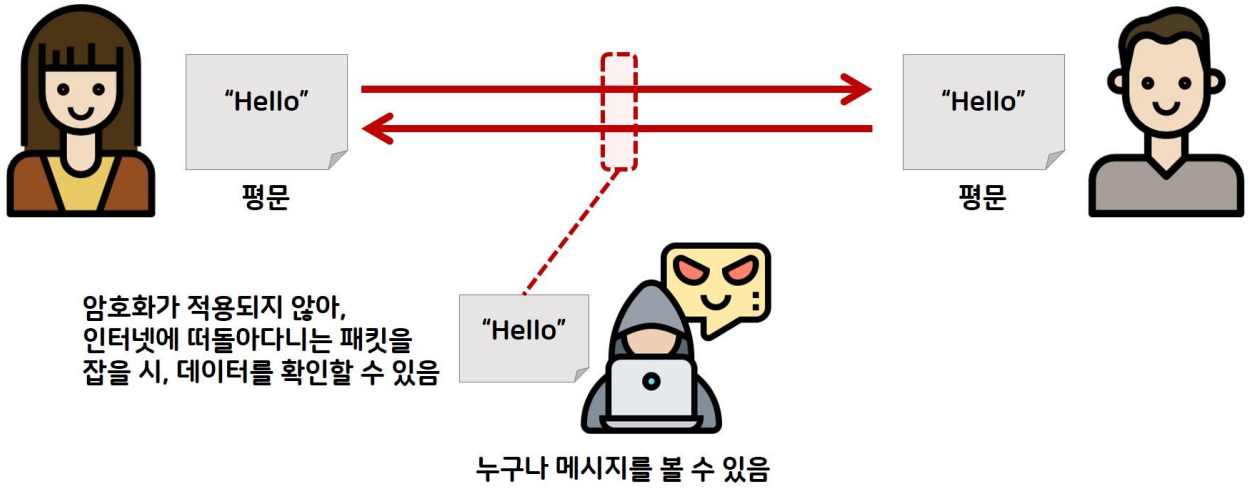
List

- Http 정의
- Https 정의 / 원리
- 브라우저가 https 웹 사이트를 믿는 과정
- Https에서 암호화 되는 정보와 암호화 되지 않는 정보
- 장점 / 단점

01. HTTP

: 하이퍼 텍스트 전송 프로토콜(Hypertext Transfer Protocol)

- **서로 다른 시스템들 사이에서 통신을 주고받게 해주는 가장 기초적인 프로토콜**
- 인터넷의 초기에 모든 웹사이트에서 기본적으로 사용되었던 프로토콜
- 문제점: 서버에서부터 브라우저로 전송되는 정보가 암호화되지 않는다
- 포트 80번

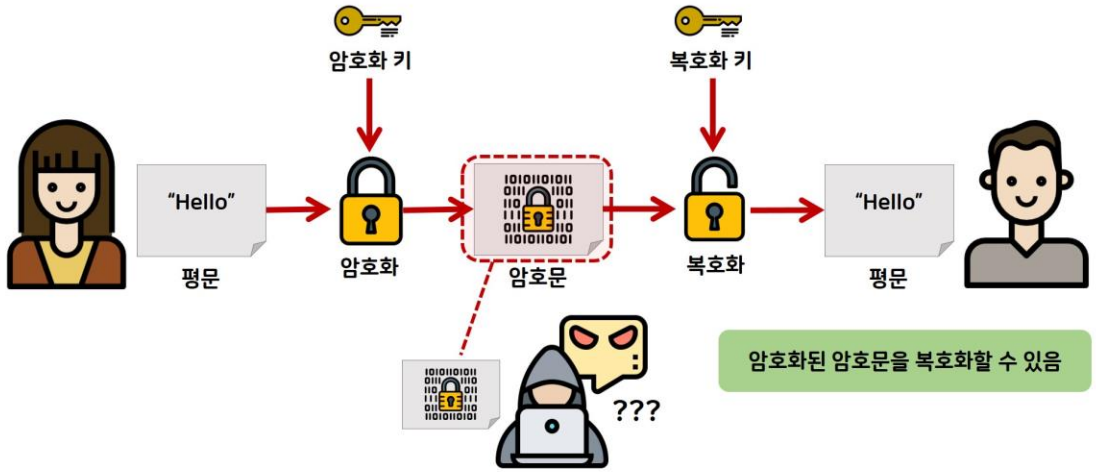


02. Https 정의 / 암호화 하는 방법

하이퍼 텍스트 전송 프로토콜 보안(Hypertext Transfer Protocol Secure)

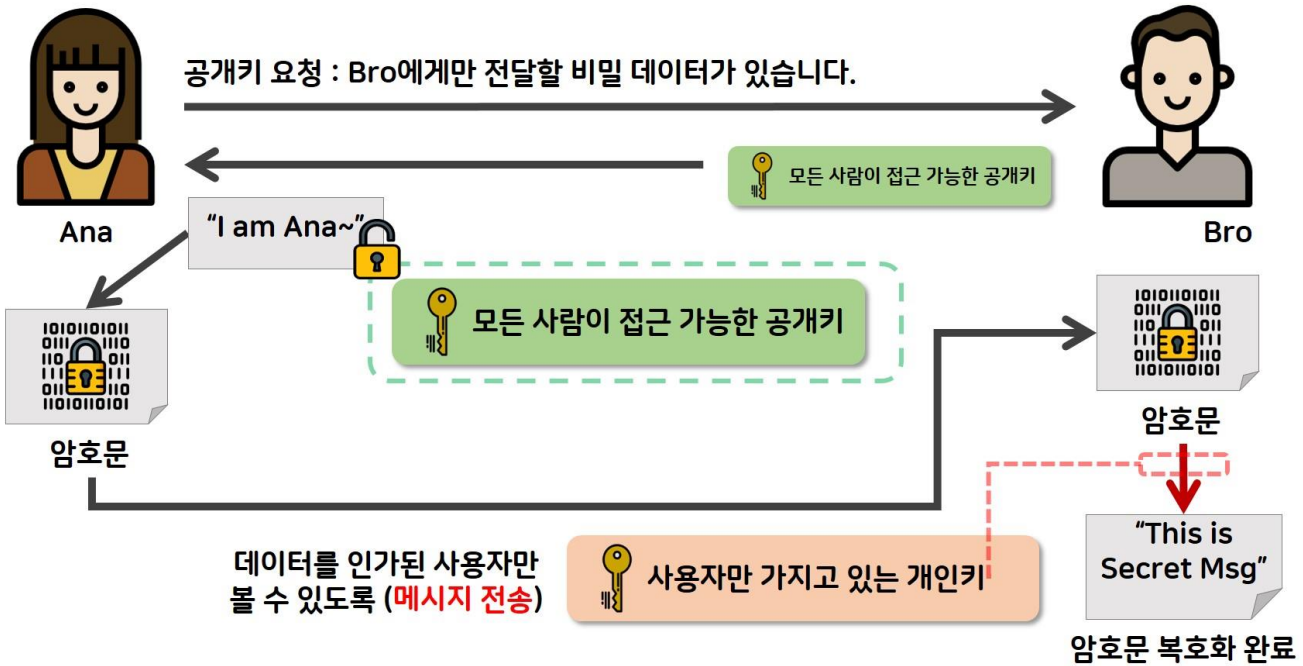
- 사용자 컴퓨터와 방문한 사이트 간에 전송되는 사용자 데이터의 무결성과 기밀성을 유지할 수 있게 해주는 인터넷 통신 프로토콜
- HTTP의 기존 소켓 대신 TLS(전송 계층 보안)을 사용

- **암호화**: 교환되는 데이터를 암호화하여 침입자로부터 보호. 즉, 사용자가 웹사이트를 탐색하는 동안 아무도 대화를 '엿들을' 수 없고 페이지에서 활동을 추적할 수 없으며 정보를 도용할 수 없습니다.
- **데이터 무결성**: 데이터가 전송되는 동안 의도적이든 그렇지 않든 모르는 사이에 데이터가 변경되거나 손상되는 일을 방지합니다.
- **인증**: 사용자가 의도된 웹사이트와 통신 중임을 입증합니다. 중간 공격을 보호하고 사용자의 신뢰를 구축하여 다른 비즈니스 이점으로 이어지게 됩니다.



02. Https 정의 / 암호화 하는 방법

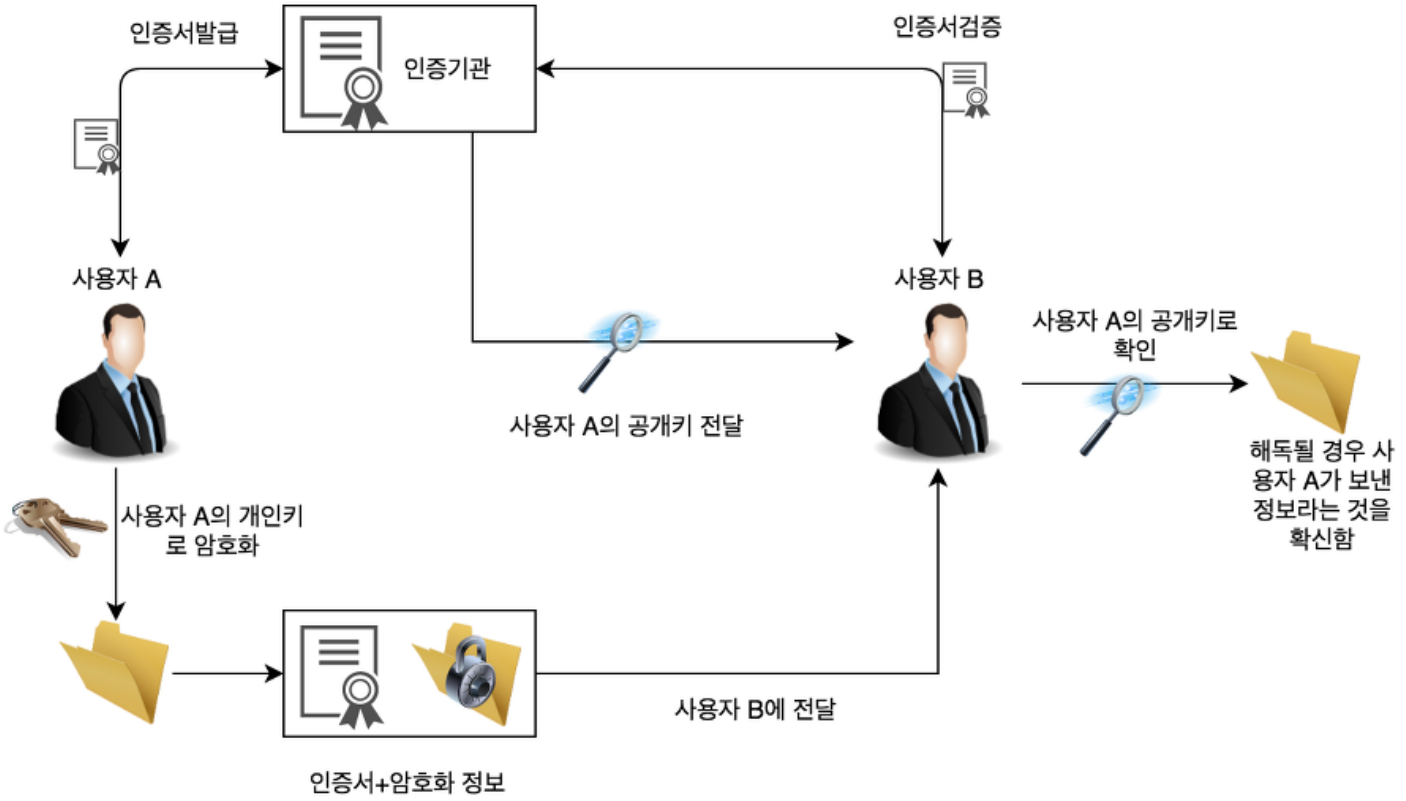
- 공개키/개인키 암호화 방식
- 공개키 암호화: 공개키로 암호화를 하면 개인키로만 복호화 → 개인키는 나만 가지고 있으므로, 나만 볼 수 있다.
- 개인키 암호화: 개인키로 암호화하면 공개키로만 복호화 → 공개키는 모두에게 공개되어 있으므로, 내가 인증한 정보임을 알려 신뢰성을 보장할 수 있다.



- 공개키로 암호화

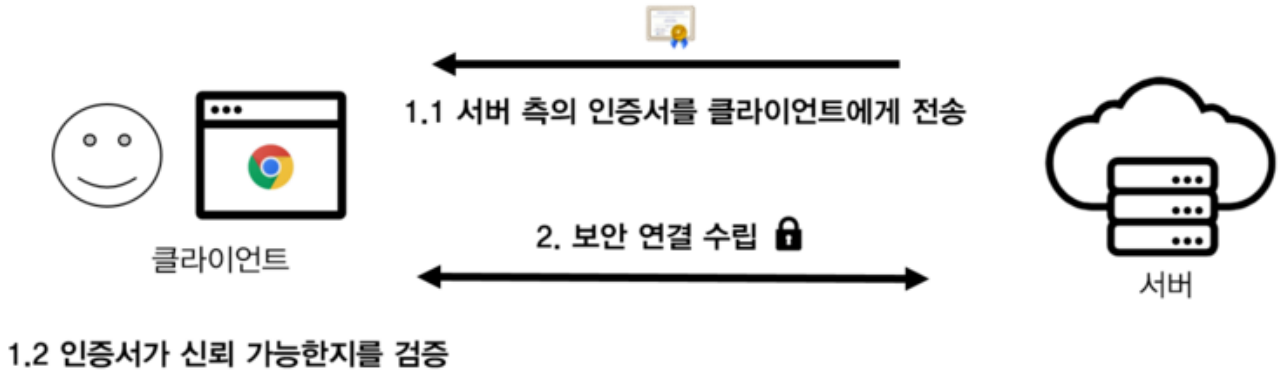
02. Https 정의 / 암호화 하는 방법

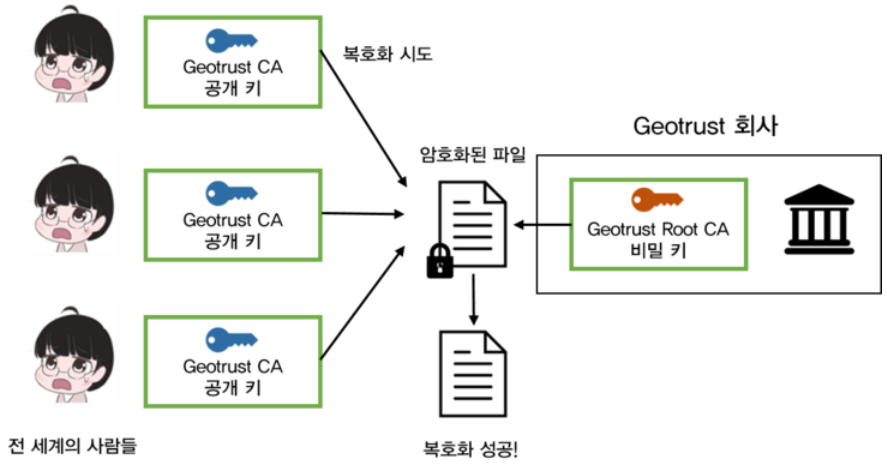
- 공개키/개인키 암호화 방식
- 공개키 암호화: 공개키로 암호화를 하면 개인키로만 복호화 → 개인키는 나만 가지고 있으므로, 나만 볼 수 있다.
- 개인키 암호화: 개인키로 암호화하면 공개키로만 복호화 → 공개키는 모두에게 공개되어 있으므로, 내가 인증한 정보임을 알려 신뢰성을 보장할 수 있다.



- 개인키로 암호화

03. 브라우저가 https 웹 사이트를 믿는 과정





[협상]

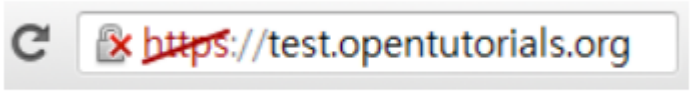
- 1단계 : 클라이언트가 서버에 어떤 암호화 방식을 사용할 것인지와 랜덤 데이터를 보냄
- 2단계: 서버가 클라이언트가 제시한 암호화 방식 중 한가지를 선택하고, 랜덤 데이터를 클라이언트에게 전달

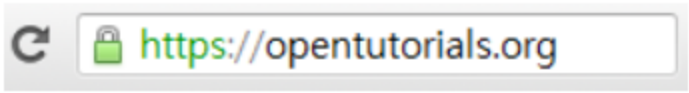
[인증]

- 3단계: 클라이언트의 CA를 통해 서버 인증서가 신뢰 가능한지를 확인 → 공개키로 복호화

[보안 연결 수립]

- 4단계 : 서버-클라이언트 간에 생성된 랜덤 값을 통해 대칭 키를 생성하고, 이를 통해 네트워크 데이터를 암호화해 전송





04. Https에서 암호화 되는 정보와 암호화 되지 않는 정보

전체 페이지 암호화 적용	부분 페이지 암호화 적용
소스 수정 간단	소스 수정 복잡
서버 부하 ↑	서버 부하 ↓

- 전체 페이지 암호화: https 프로토콜 URL으로 페이지가 열리도록 https 프로토콜 호출하는 방법, 리다이렉션(Redirection) 설정하는 방법. 소스 수정을 통해 적용이 가능하나 암호화 적용이 필요 없는 부분까지 암호화되기 때문에 부분 암호화 방식에 비해 서버에 부하를 줄 수 있음
- 부분 페이지 암호화는 암호화가 필요한 부분에 소스 수정을 하여 서버의 부하가 증가하는 것을 줄일 수 있습니다.

05. 장점 / 단점

	HTTP	HTTPS
장점	HTTPS보단 빠름	안전하게 데이터를 주고 받음. 신뢰성 보장 가속화된 모바일 페이지 (AMP, Accelerated Mobile Pages)를 만들고 싶을 때 사용
단점	보안에 취약	인증서 발급 추가 비용 발생 속도가 느리다 (암호화 /복호화 과정)

* AMP란 모바일 기기에서 훨씬 빠르게 콘텐츠를 로딩 하기 위한 방법으로 구글이 만든 것

참고 사이트

http://blog.wishket.com/http-vs-https-%EC%B0%A8%EC%9D%B4-%EC%95%8C%EB%A9%B4-%EC%82%AC%EC%9D%B4%ED%8A%B8%EC%9D%98-%EB%A0%88%EB%B2%A8%EC%9D%B4-%EB%B3%B4%EC%9D%B8%EB%8B%A4/
https://mangkyu.tistory.com/98
https://velog.io/@bclef25/http%EC%99%80-https%EC%9D%98-%EC%B0%A8%EC%9D%B4
https://cheese10yun.github.io/https/
https://opentutorials.org/course/228/4894
https://developers.google.com/search/docs/advanced/security/https?hl=ko
https://m.blog.naver.com/alice_k106/221468341565
https://raonctf.com/essential/study/web/asymmetric_key
http://yii.inet.co.kr/security/ssl03.php