TLS

송민호

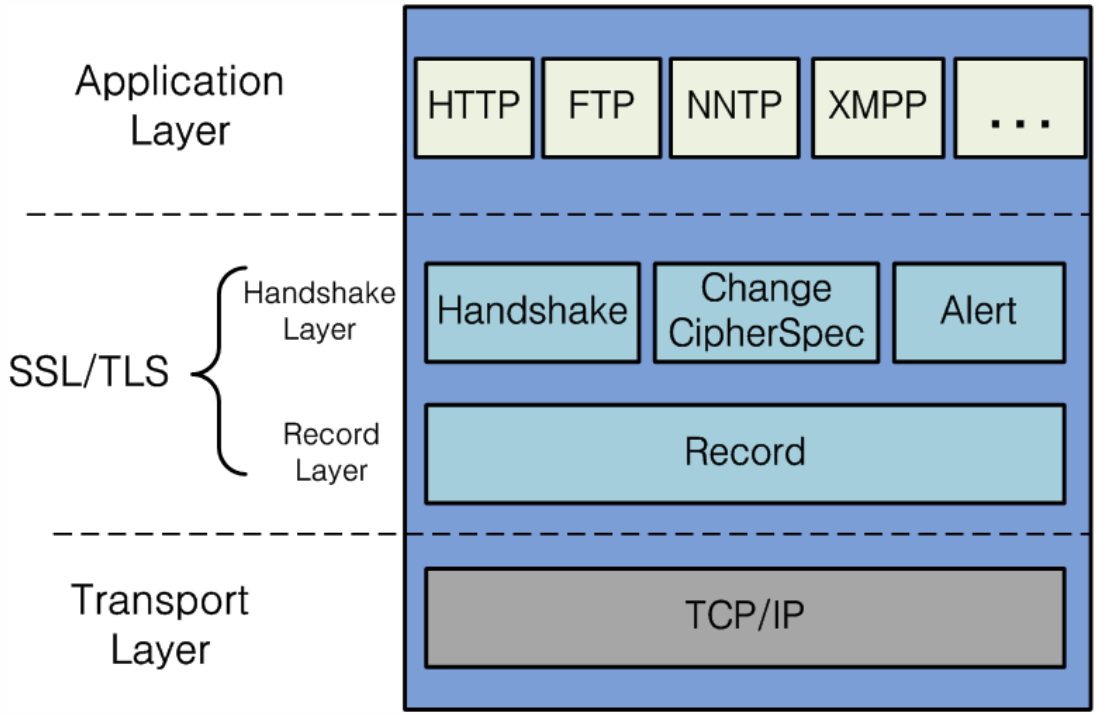
유튜브 주소 : https://youtu.be/ztU8f4TLmSM

TLS

- TLS(Transport Layer Security)
  - 인터넷 상에서 통신할 때 주고받는 데이터를 보호하기 위한 표준화된 **암호화 프로토콜**
    - 인증(Authentication), 암호화(Encryption), 무결성(Integrity) 제공
  - Netscape에 의해 개발된 SSL(Secure Socket Layer) 3.0버전을 기반으로 하며 현재 최종 버전은 2018년 8월에 발표된 TLS 버전 1.3이다.
  - TLS는 전송계층(Transport Layer)의 암호화 방식이기에 HTTP, FTP, XMPP 등 응용계층(Application Layer)프로토콜의 종류에 상관없이 사용이 가능.

TLS

- Handshake
  - 서버와 클라이언트가 **상호 인증**하는 과정
- ChangeCipherSpec
  - 암호 통신을 위한 **보안 알고리즘 정보의 결정**과 **보안 파라미터를 상대에게 전송**하는데 사용
  - 현재 작동되는 보안 파라미터 변경
- Alert
  - Handshake 과정에서 오류 발생시 상대방에게 **오류 통보** 기능 수행
    - Ex) 상대방이 제시한 암호화 방식을 지원할 수 없을 때 등
- Record
  - 상위 프로토콜 및 Application 메시지를 수납하여 레코드 단위로 운반하면서 **주요 기능**을 제공
    - 주요 기능 – 메시지 분할, 압축, 무결성, 인증, 암호화



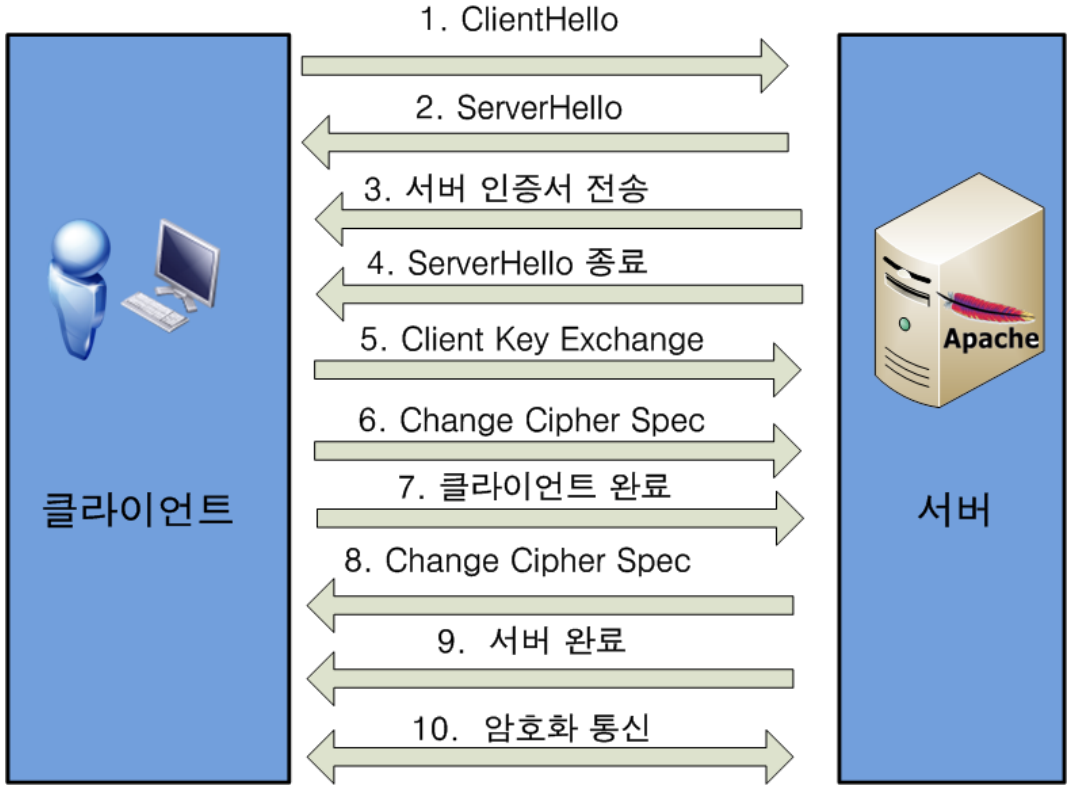
TLS Handshake

- **Handshake** 과정을 거친 후에 구축
  - Handshake – 데이터를 전송하기 전에 먼저 정확한 전송을 보장하기 위해 **상대방 컴퓨터와 사전에 세션을 수립하는 과정**

  - 클라이언트와 서버는 Hello 메시지로 기본적인 정보를 송수신 (1, 2)

  - 서버는 서버가 사용하는 SSL/TLS 인증서를 전달 (3, 4)

  - Key Exchange 과정 : 클라이언트는 암호화 통신에 사용할 대칭키를 생성하고 서버에 전달 (5) 주로 RSA나 디피-헬만 키 교환(Diffie-Hellman key exchange)을 많이 사용함.

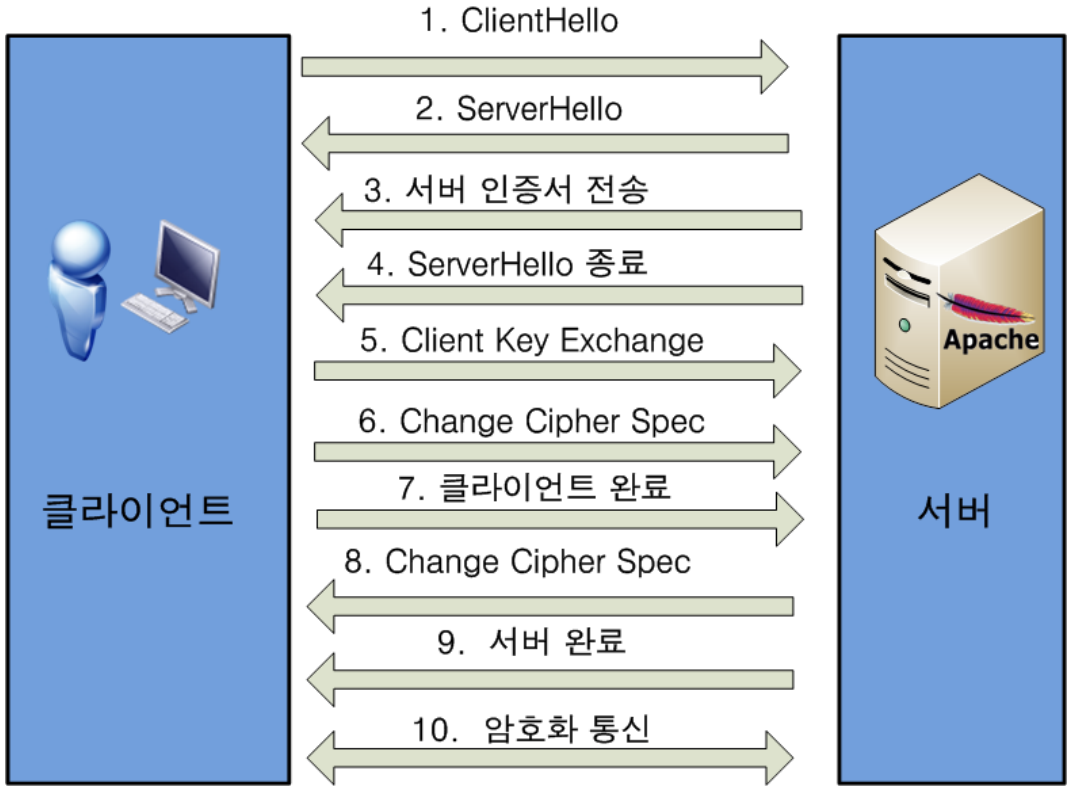


TLS Handshake

- 클라이언트는 암호화 통신에 사용이 가능한 암호 알고리즘과 해시 알고리즘 목록을 서버에 전달 (6, 7)

- 서버도 알고리즘 목록을 교환 후 Handshake가 종료되며 클라이언트와 서버는 암호화 통신에 필요한 대칭키를 서로 보유 (8, 9)

➢ 모든 과정이 끝나면 SSL(TLS) 세션이 구축 실제 암호와 통신 시작 가능 (10)



TLS 버전

- SSL v1/2/3
  - 1.0 버전은 치명적인 보안 결함 때문에 공개된 적이 없음
  - 2.0 버전은 보안 취약점이 발견되어 다음 해 3.0 버전으로 대체
  - 3.0 버전도 취약점(POODLE, DROWN)이 발견됨

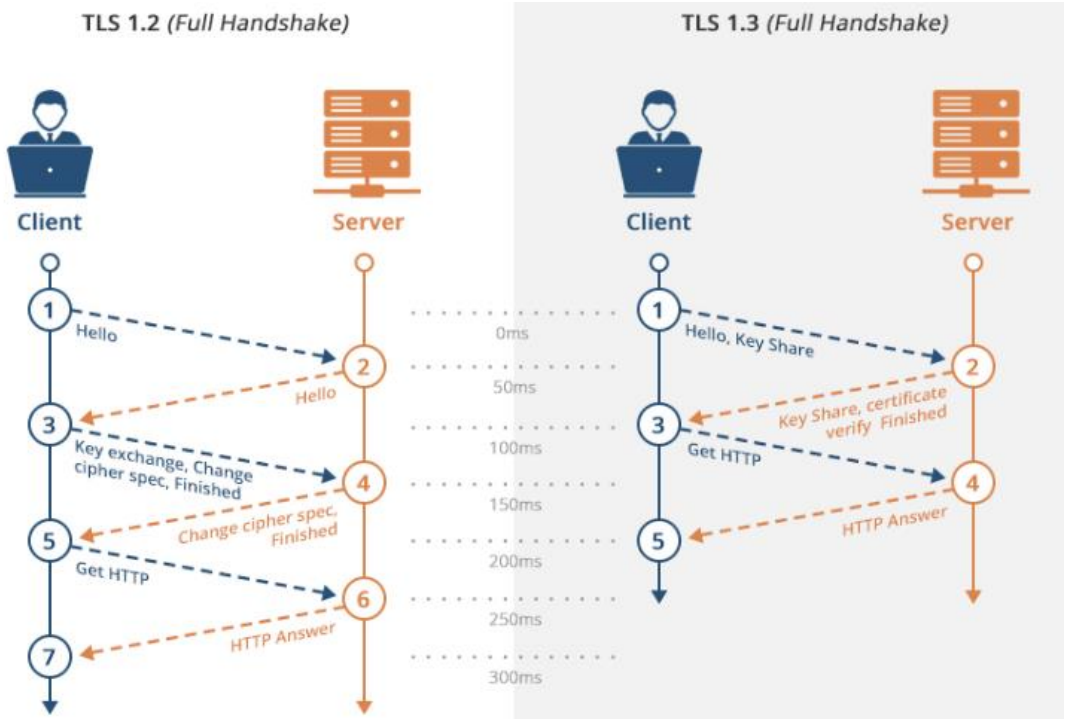
- TLS 1.0
  - SSL 3.0의 업그레이드 버전, SSL 3.0의 대부분의 취약점을 해결
  - 주로 SHA1 알고리즘 사용, SHA2도 지원
- TLS 1.1
  - 블록체인 공격에 대한 방어와 IANA 등록 파라미터 지원 추가
- TLS 1.2
  - 취약한 SHA1알고리즘 사용 중단, SHA2 사용으로 변경

TLS 1.3

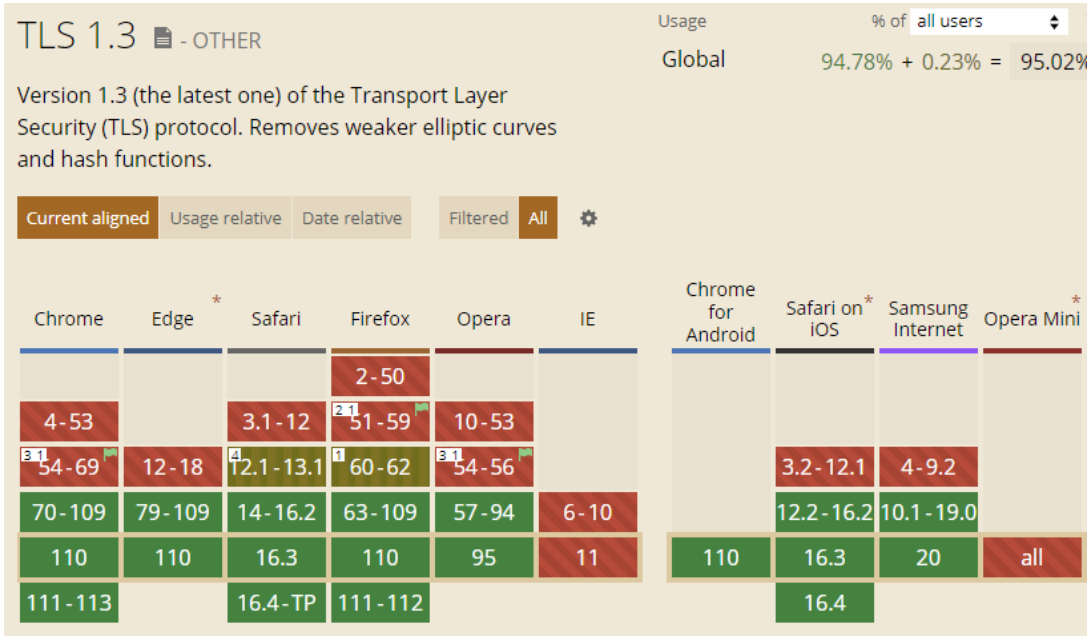
- 최초 연결 시 암호화 통신을 개시하는 절차 간소화(왕복 2회 -> 1회)
- 오래된 안전하지 않은 알고리즘 제거(SHA1, MD5, DES 등)
- 0-RTT(Zero Round Trip Time Resolution) 기능으로 연결 속도 개선
  - 이전에 방문한 웹 사이트에 대한 연결 속도 증가

버전	연결 종류	TCP HandShake	TLS HandShake	HTTP	합
TLS 1.2 이전	새 연결	1 RTT	2 RTT	1 RTT	4 RTT
	재개된 연결	1 RTT	1 RTT	1 RTT	3 RTT
TLS 1.3	새 연결	1 RTT	1 RTT	1 RTT	3 RTT
	재개된 연결	1 RTT	1 RTT	1 RTT	3 RTT
TLS 1.3+0-RTT	새 연결	1 RTT	1 RTT	1 RTT	3 RTT
	재개된 연결	1 RTT	0 RTT	1 RTT	2 RTT

TLS 1.3



암호화 통신 간소화 과정



2023-03-08
TLS 1.3 browser support

TLS 라이브러리 - OpenSSL

- SSL 및 TLS 프로토콜을 사용하기 위한 오픈 소스
  - C언어로 작성된 중심 라이브러리 안에 기본적인 암호화 기능 및 여러 유틸리티 함수들이 구현되어 있음

- OS 지원
  - 윈도우, OpenVMS, 유닉스 계열(맥 OS, 리눅스 등)

OpenSSL
Cryptography and SSL/TLS Toolkit

- 암호화 알고리즘 지원
  - 암호문 – AES, blowfish, RC4, DES 등
  - 해시 함수 – MD5, SHA1, SHA2 등
  - 공개 키 암호 방식 – RSA, DSA, Diffie-Hellman 등

TLS 라이브러리 - WolfSSL

- 임베디드 시스템 사용을 타겟으로 한 소형 SSL/TLS 라이브러리
  - OpenSSL 경량화 버전
  - C언어로 작성된 TLS의 오픈 소스를 구현
  - C, python등의 언어 지원

- OS 지원
  - 윈도우, Linux, macOS, Android 등

- 암호화 알고리즘 지원
  - 암호문 – AES, IDEA, NTRU 등
  - 해시 함수 – MD5, SHA1, SHA2 등
  - 공개 키 암호 방식 – RSA, Diffie-Hellman 등

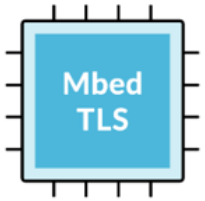
TLS 라이브러리 - mbedTLS

- SSL, TLS와 각각의 암호화 알고리즘 및 필요한 지원코드를 구현
  - 소형 임베디드 장치에 맞도록 설계
  - 핵심 SSL 라이브러리는 C언어로 작성됨

- 대부분의 플랫폼에서 사용 가능
  - Linux, 윈도우, Android, iOS 등
  - ARM, x86 아키텍처 등

Mbed-TLS/**mbedtls**

An open source, portable, easy to use, readable and flexible TLS library, and reference implementation of the PSA Cryptography API.



- 암호화 알고리즘 지원
  - 암호문 – AES, ARIA, blowfish, DES 등
  - 해시 함수 – MD5, SHA1, SHA2 등
  - 공개 키 암호 방식 – RSA, Diffie-Hellman, ECC 등

Q & A