SSL/TLS Study log

2017-11-24

양혜임

- SSL/TLS : 응용계층과 전송계층 사이에서 보안조치를 취하는 웹 보안 방법 중 하나
  전송 계층의 프로토콜인 TCP 바로 위에서 보안을 구현하는 것

기존의 통신 방식

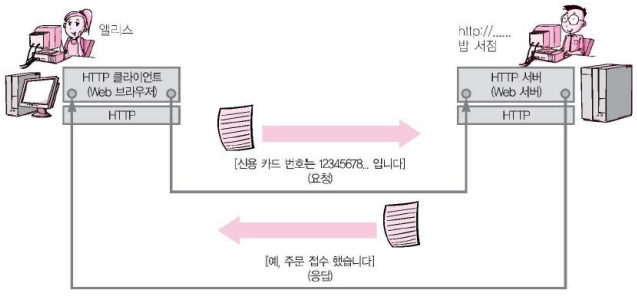

그림 14-2 SSL/TLS를 사용하지 않고 신용카드 번호를 보냈을 경우

SSL/TLS를 이용한 통신 방식

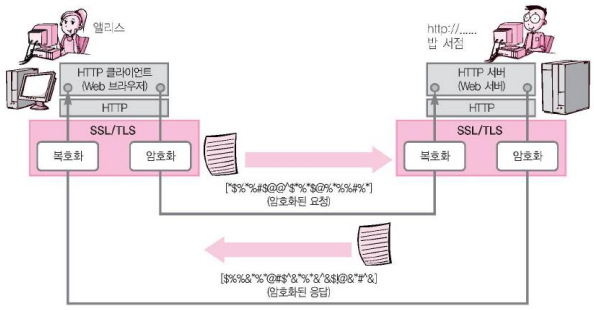

그림 14-3 HTTP를 SSL/TLS 상에 올려서 요청과 응답을 암호화한다

- 통신 내용을 암호화해 주는 프로토콜로서 SSL 혹은 TLS를 이용,
- SSL/TLS 상에 HTTP를 올림. (프로토콜의 이중 구조)
- 요청과 응답을 암호화

- 암호 스위트 (Cipher Suite)

SSL/TLS에서 사용하는 대칭 암호, 공개 키 암호, 디지털 서명, 일방향 해시 함수 등은 부품과 같이 교환 가능
사용하고 있던 암호 기술에 결함이 발견되었을 때는 그 부분을 교체할 수 있음 (모듈화된 프레임워크)
암호 기술의 '추천 세트 ' 가 SSL/TLS로 규정되어 있으며 이 추천 세트를 암호 스위트라고 함.

- SSL vs TLS

SSL (Secure Socket Layer)
1994년 Netscape사에 의해 만들어 짐
많은 웹 브라우저에서 사용되어 사실상 업계 표준이 됨
TLS (Transport Layer Security)
SSL 3.0을 기초로 해서 IETF가 만든 프로토콜
TLS 1.0은 SSL 3.1에 해당되는 것

- 계층화된 프로토콜

구성

- TLS 레코드 프로토콜

암호화 처리

- TLS 핸드셰이크 프로토콜

암호화 이외의 다양한 처리 수행
4개의 서브 프로토콜로 나누어짐

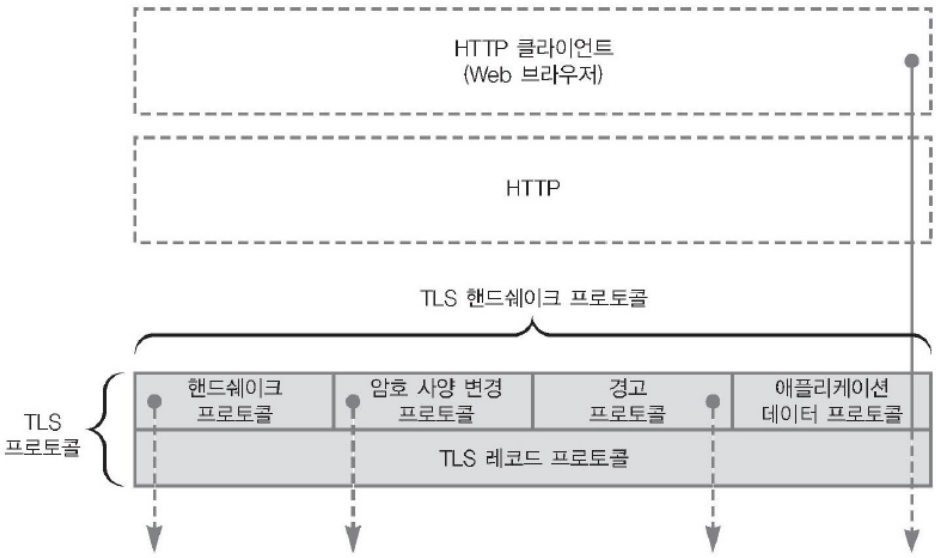

그림 14-5 TLS 프로토콜 계층

SSL/TLS Study log # 2017-11-24

양혜임

▶ TLS 레코드 프로토콜

<< 메시지의 압축, 암호화, 데이터의 인증 >>

- TLS 핸드셰이크 프로토콜 밑에 있으며, 대칭 암호를 사용해서 메시지를 암호화하고 통신하는 부분
- TLS 레코드 프로토콜에서는 대칭 암호와 메시지 인증 코드를 이용하지만, 알고리즘과 공유 키는 핸드셰이크 프로토콜을 사용해서 서버와 클라이언트가 상담하여 결정

그림 14-6 TLS 레코드 프로토콜의 처리

▶ TLS 핸드셰이크 프로토콜

- 핸드셰이크 프로토콜

  << 공유 키 생성(암호통신) 및 인증서 교환(인증) >>

  클라이언트와 서버가 암호 통신에 사용할 알고리즘과 공유 키를 결정하기 위한 것
  인증서를 이용한 인증도 여기에서 수행
  4개의 서브 프로토콜 중 가장 복잡

- 암호 사양 변경 프로토콜

  TLS 핸드셰이크 프로토콜의 일부
  암호 방법을 변경하는 신호를 통신 상대에게 전하기 위한 것

- 경고 프로토콜

  에러가 발생했다는 것을 통신 상대에게 전하기 위한 것

- 애플리케이션 데이터 프로토콜

  TLS 상에 올라 있는 애플리케이션의 데이터를 통신 상대에게 전하기 위한 것

▶ 절차

		보내지는 정보 (주요 정보만 작성)
1. ClientHello	(C → S)	사용가능한 버전번호, 암호 스위트, 압축 방법 목록 등
2. ServerHello	(S → C)	사용하는 버전 번호, 암호 스위트, 압축 방법 등
3. Certificate	(S → C)	인증서 목록
4. ServerKeyExchange	(S → C)	
5. CertificateRequest	(S → C)	서버가 이해할 수 있는 인증서 타입, 인증기관 이름 목록
6. ServerHelloDone	(S → C)	
7. Certificate	(C → S)	
8. ClientKeyExchange	(C → S)	암호화한 사전 마스터 비밀
9. CertificateVerify	(C → S)	
10. ChangeCipherSpec	(C → S)	
11. Finished	(C → S)	
12. ChangeCipherSpec	(S → C)	
13. Finished	(S → C)	

14. 애플리케이션 데이터 프로토콜로 이행