https/http와 공인인증서

김예찬

1. http / https

1) http (hypertext transfer protocol)

인터넷에서 하이퍼텍스트(hypertext) 문서를 교환하기 위하여 사용되는 통신규약이다. 하이퍼텍스트는 문서 중간중간에 특정 키워드를 두고 문자나 그림 을 상호 유기적으로 결합하여 연결시킴으로써, 서로 다른 문서라 할지라도 하나의 문서인 것처럼 보이면서 참조하기 쉽도록 하는 방식을 의미한다.

http는 1989년 팀 버너스 리(Tim Berners Lee)에 의하여 처음 설계되어 인터넷을 통한 월드 와이드 웹(World-Wide Web) 기반에서 전 세계적인 정보공유를 이루는데 큰 역할을 하였다. http의 첫번째 버전은 인터넷을 통하여 가공되지 않은 데이터를 전송하기 위한 단순한 프로토콜이었으나, 데이터에 대 한 전송과 요구·응답에 대한 수정 등 가공된 정보를 포함하는 프로토콜로 개선되었다.

인터넷 주소를 지정할 때 'http://www....'와 같이 하는 것은 www로 시작되는 인터넷 주소에서 하이퍼텍스트 문서의 교환을 http 통신규약으로 처리하라 는 뜻이다.

2) https (Secure Hypertext Transfer Protocol)

https 란 인터넷 상에서 정보를 암호화하는 SSL(Secure Socket Layer) 프로토콜을 이용하여 데이터를 전송하고 있다는 것을 의미한다. SSL 프로토콜은 테리사(Terrsa)가 개발해 Netscape사가 NetSite의 암호화 중심 프로토콜로 정착시킨 기술로 정보 암호화시 공개키(Pubilc Key)와 개인키(Private Key)라는 두가지 키를 이용하는 방법이다.

공개키(Public Key)는 인터넷 상에 공개되어 있는 키로 서버에서 자동으로 이용자님의 브라우져로 보내며, 이용자가 입력한 정보는 이 공개키(Public Key)를 이용하여 암호화되어 다시 서버까지 전달된다. 이 공개키(Public Key)로 암호화된 정보는 서버만이 유일하게 소유하고 있는 개인키(Private Key)로만 해독이 가능하다. 제3자가 의도적으로 암호화된 이용자 정보를 가로챈다 하더라도, 서버의 개인키(Private Key)가 없는 한은 해석이 불가능하다.

2. 공인인증서

1) 국내에 공인인증서가 생긴 배경과 그 위험성은?

대한민국에서는 1999년 전자서명법을 제정하고 공인인증서의 발급 및 관리의 체계를

마련하였다. 행정안전부 장관이 지정한 공인인증기관에서 발급한 인증서를 공인인증서라 하며 기타 인증기관에서 발급한 인증서를 사설인증서라 부른다.

공인인증서가 사용되는 분야는 인터넷뱅킹·증권거래·인터넷을 통한 카드 결제·보험 등의 금융업무와 전자세금계산서·전자입찰·전자계약 등의 기업 조달업무, 정부에서 제공하는 전자민원·전자정부 업무 등이 있다.

2) 공인인증서의 문제점

사실 공인인증서의 문제점은 공인인증서의 배포 당시 윈도우 운영체제의 다양한 취약점, 그리고 스마트카드 리더(reader)의 사용에 대해 반대했던 금융권의 설계 원칙에서부터 시작됐다. 공인인증서의 문제점은 대체로 암호학적 문제, 액티브X 문제, 운영체제 문제가 대표적이다.

- 암호학적 문제점: 설계부터 잘못된 시스템

대부분의 정보보호 교과서에서는 비밀키는 안전하게 저장돼야 한다고 이야기를 하고 그 방법으로 하드웨어에 저장하며 비밀키를 이용한 연산은 하드웨어 내부에서 일어나야 한다고 이야기한다. 교과서의 이런 가르침과 달리 우리나라의 공인인증서는 처음부터 NPKI 디렉토리에 사용자의 비밀키로 암호화되어 저장되어 왔다. 이런 설계로 인해 공인인증서의 안전성은 전적으로 메모리 해킹의 가능성 여부와 비밀번호의 안전성에 의존하게 됐으며, 공개키의 안전성과는 동떨어지게 됐다. 즉, 국내 공인인증서는 암호학적으로 처음부터 잘못 설계된 시스템이다.

- 액티브X 문제점: 사용자 UX의 관성

액티브X(ActiveX) 또한 설계 초기부터 많은 논란을 가져왔었고 지금도 논란 속에 있다. 이런 논란의 핵심은 크게 두 가지로 나눠 볼 수 있다.

먼저 액티브X 자체의 취약점이다. 액티브X의 취약점이 있을 경우 공인인증서의 비밀키가 노출될 위험이 있다고 주장한다. 모든 사용자가 서드파티 애플리케이션 프로그램을 설치하겠냐는 질문을 윈도우 운영체제에서 물어볼 때 확인을 누르는 '관성'이 생긴다는 점이다. 끊임없이 새로운 소프트웨어를 깔라고 강요하고 새로운 사이트에 들어갈 때마다 새로운 소프트웨어들을 설치하라고 강요함으로써 사용자들은 '확인' 버튼을 누르는 것을 매우 자연스럽게 생각하게 되었다.

즉, 하나의 소프트웨어를 다양한 애플리케이션에 적용할 수 있지 않고 모든 애플리케이션마다 다른 소프트웨어를 깔아야 하는 것이 문제의 핵심인 것이다. 이런 소프트웨어의 숫자가 늘어나면 취약점은 늘어날 수 밖에 없고 취약점이 늘어나면 다양한 공격 또한 가능하다.

- 운영체제의 문제점: 윈도우와 안드로이드 자체의 많은 취약점

공인인증서 개발 초기부터 가장 많이 사용되어 왔던 윈도우, 그리고 최근에 점차 사용이 늘어가고 있는 안드로이드, 이 두 운영체제는 많은 취약점을 노출해 왔다. 먼저 안드로이드는 루팅(rooting)을 허용해야 하는 시스템이다. 또한 웹키트(webkit), 브라우저 등 다양한 취약점이 계속 존재해 왔고 ASRL(address space layout randomization) 또한 최근에 도입이 됐지만 여전히 취약점은 존재하고 있다.

뿐만 아니라 안드로이드 운영체제 위에 개발되는 애플리케이션들의 취약점은 이런 문제를 악화시키고 있다. 윈도우는 최근 보안이 과거에 비해 많이 강화됐지만 여전히 새로운 운영체제의 도입이 지연되고 있으며, 특히 우리나라의 다양한 공인인증용 소프트웨어는 보안성을 약화시킨다.

여기에 앞에서 언급한 액티브X의 취약점, 그리고 그 기반 위에 개발되는 공인인증용 소프트웨어의 취약점들은 다양한 개발 과정 상에서 완벽한 구현을 요구하고 있다.따라서 현재 운영체제의 안전성에 근간을 둔 공인인증용 소프트웨어 개발은 원천적인 문제를 갖고 있다. 마지막으로 루팅이 될 경우 최근 이슈가 되고 있는 메모리 해킹, 그리고 키보드 로깅 등은 전문적 해커에게는 어려운 기술이 아니다.

3. 위 내용을 조사하며 느낀점

미국에 있을 때 시티은행과 체이스 은행을 쓰고 아마존을 자주 이용했다. 처음에 한국에서 미국으로 건너가기 전에는 엑티스 엑스와 공인인증서가 귀찮다고 생각했지 보안, 관리 보수, 이용면에서 크게 문제가 있는지 몰랐다. 하지만 미국에서 다음과 같은 서비스를 이용하니 너무 편한 것을 알게 되었다. 카드번호만 입력하고 다른 소프트웨어를 깔지 않아도 되는 상황이 있었고 은행계좌와 모바일 이체가 자유로웠다.

그런 편의를 누리다 한국에 돌아오니 불편한 점이 이만저만이 아니다. 은행공인 인증서, 쇼핑몰 결제까지 어떤 걸 해도 짜증이 난다. 그리고 개발을 배우는 입장에서 좋은 기술을 고민하다 보니 *http* 와 *https* 프로토콜을 알게 되었다. 좋은 기술을 개발하는 것도 중요하지만 그것에 대한 중요성 인식과 환경조성도 얼마나 중요한지 알게 되었다.