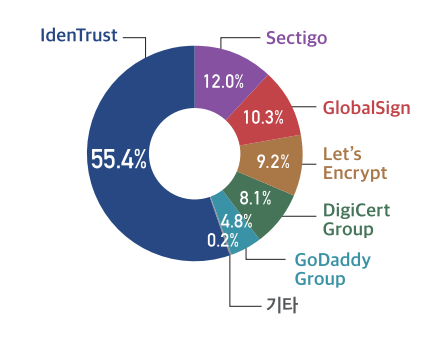
인터넷에서 사용되는 여러 기술 : HTTPS 3

조인준
KBS 미디어기술연구소 차장

HTTPS(Hypertext Transfer Protocol Secure)의 구현과 동작에는 서버, 클라이언트, 인증기관(CA, Certificate Authority)의 세 요소가 필수입니다. 우리가 인터넷을 이용할 때 클라이언트를 통해 서버에 정보나 서비스를 요청하고 서버는 클라이언트가 요청한 사항을 처리하며, 이에 대한 회신을 주므로 서버와 클라이언트는 익숙한 개념인데 인증기관은 서버나 클라이언트처럼 우리가 직접적으로 접하는 대상이 아니므로 그것이 무엇인지에 관해 특별히 신경을 쓰지 않게 되는 것 중의 하나입니다. 하지만 HTTPS가 어떻게 동작하는지를 알기 위해서는 인증기관 및 인증기관이 발급한 인증서에 대해 아는 것이 필요하므로 인증기관에 대한 설명을 드리겠습니다.

인증기관	점유율
IdenTrust	55.4%
Sectigo	12.0%
GlobalSign	10.3%
Let's Encrypt	9.2%
DigiCert Group	8,1%
GoDaddy Group	4.8%
기타	0.2%

표 1. 인증기관 및 기관별 점유율



인증기관은 공개키 기반 구조(Public Key Infrastructure)의 일부로서 HTTPS 구현에 필요한 인터넷 보안 기술인 SSL/TSL에 사용되는 인증서를 발급합니다. 인증기관은 서버와 클라이언트 모두가 신뢰합니다. 이 신뢰를 기반으로 인증기관의 인증서를 가진 서버나 클라이언트 또한 신뢰할 수 있습니다. 결국 인터넷 등의 컴퓨터 네트워크를 통해 서버와 클라이언트 사이에 정보 및 서비스 관련 거래가 이루어질 때 서버는 인증서를 통해 자신이 신뢰할 수 있음을 클라이언트에게 증명하고, 이 인증서를 믿고 클라이언트는 서버와 보안이 제공되는 환경에서 데이터를 주고받습니다.

웹에서 사용되는 기술과 트렌드에 관한 통계를 수집하고 제공하는 W3Techs(w3techs.com)에 따른 인증기관들의 현재 시장 점유율은 [표 1]과 같습니다.

그림 1. 인증기관과 자체 서명 인증서

앞서 언급했듯이 HTTPS 보안의 세 중요 요소 중에서 인증기관은 신뢰의 초석을 제공하는 역할을 합니다. 인증기관도 암호화 및 서명에 필요한 사설키와 공용키를 가지고 있으며 [그림 1]과 같이 자신의 사설키로 직접 서명한 자체 서명 인증서에는 자신의 공용키가 포함되어 있습니다. 이전에 설명해 드렸듯이 사설키로 암호화된 데이터는 이에 매칭된 공용키로밖에 복구가 안 되고, 공용키로 암호화된 데이터는 이에 매칭된 사설키로밖에 복구가 안 됩니다. 따라서 자신의 사설키로 암호화한 것은 자신이 외부에 공개한 공용키로만 복구 가능하므로 이 공용키로 복구해서 정상적인 데이터가 확인되면 이 데이터는 매칭되는 사설키 소유자가 만든 것이 틀림없다는 증명이 되므로 자신의 사설키로 어떤 데이터를 암호화하면 그것이 그 데이터에 대한 자신의 서명이 되는 것입니다.

그림 2. 인증서 요청

어떤 기관에서 자신의 서버에 인증기관이 발급한 인증서를 설치하여 HTTPS 기반의 보안을 제공하고자 한다면 이 기관은 우선 자신의 사설키와 공용키를 생성한 후에 [그림 2]와 같이 인증서 발급에 필요한 정보를 담은 CSR(Certificate Signing Request)을 인증기관에 보냅니다. CSR은 실질적으로는 인증서 발급에 필요한 정보를 담은 파일에 해당하며 신청한 기관의 공용키와 기관 서버의 도메인 정보 등이 포함됩니다. 중요한 것은 CSR은 인증서를 신청한 서버의 사설키로 서명이 되어 있다는 것입니다. CSR을 받으면 인증기관은 이를 검증하여 인증서를 신청한 기관의 서버가 안전하고 신뢰할 수 있는지를 판단합니다.

그림 3. 인증서 발행

CSR을 검토하여 안전하고 신뢰할 수 있는 기관의 서버라고 판단되면 CSR을 통해 받은 요청 기관의 공용키가 포함된 인증서를 자신의 사설키로 서명하여 발급합니다. 인증기관의 사설키로 서명된 인증서는 인증기관의 공용키로만 복원이 가능합니다. 이렇게 발급받은 인증서를 클라이언트에게 제공함으로써 서버는 자신이 안전하고 믿을 수 있다는 것을 인정받을 수 있습니다.

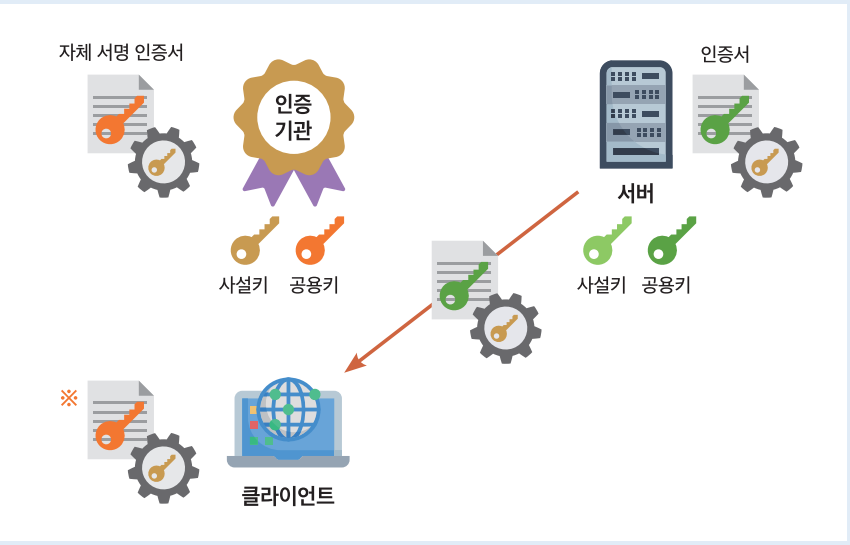


그림 4. HTTPS 연결을 위한 클라이언트의 인증서 확인

이제 클라이언트와 서버가 어떻게 인증서를 사용하여 HTTPS 기반의 보안 연결을 수립하는지에 관해 알아보겠습니다. HTTPS를 통한 보안 연결의 가장 흔한 형태는 [그림 4]의 예와 같은 브라우저(클라이언트)를 통한 웹서비스 이용입니다. 브라우저는 보안 연결을 수립하기 위해 서버로부터 인증서를 요청하여 받습니다. 서버로부터 받은 인증서는 서버 자신의 공용키를 포함하고 있고, 인증기관의 사설키로 서명되어 있습니다. 인증서를 받으면 클라이언트인 브라우저는 2가지 확인을 하게 됩니다.

① 서버 인증서가 인증기관에 의해 발행된 것이 맞는지 여부
② 서버 인증서의 소유자가 서버가 맞는지 여부

우선 ①번 서버 인증서가 인증기관에 의해 발행된 것이 맞는지에 대한 확인은 서버 인증서의 서명(인증기관의 사설키로 서명)을 인증기관의 공용키를 통해 확인하는 것으로 가능합니다. [그림 4]의 클라이언트 좌상단에 ※로 표시된 것과 같이 대부분의 브라우저는 인증기관의 자체 서명 인증서를 가지고 있어서 이 자체 서명 인증서에 포함된 공용키를 통해 서버 인증서의 서명(인증기관의 사설키로 서명)을 확인합니다.

서버의 인증서가 인증기관에 의해 발행된 것이 맞으면 그다음은 ②번 서버 인증서의 소유자가 서버가 맞는지를 확인할 차례입니다. 이를 확인하는 이유는 서버의 인증서는 공개된 것이므로 누군가가 이를 이용하여 마치 자기가 해당 서버인 것처럼 클라이언트를 속일 수 있기 때문입니다. 이를 위해 일련의 절차를 거쳐 클라이언트는 이전에 설명된 대칭키(많은 양의 데이터를 주고받을 때 계산 부담을 줄이기 위한 것으로 이전에 설명) 생성에 필요한 정보를 만들고 이를 서버가 인증서에 포함하여 보내준 공용키로 암호화하여 서버에 전달합니다. 서버의 인증서는 ①번을 통해 인증기관이 발급한 것이 맞고, 중간에 위변조가 없었다는 것이 증명되었으므로 인증서에 포함된 공개키는 조작될 수 없습니다. 조작되지 않은 서버의 공용키로 암호화된 데이터는 서버의 사설키로 밖에 복구가 안 되므로 서버가 클라이언트가 보낸 정보를 제대로 복원하여 양쪽이 모두 같은 대칭키로 정상적인 통신을 한다면 인증서를 보낸 서버가 진짜 서버인 것이 확인됩니다. 만약 반대로 서버가 클라이언트가 보낸 정보를 제대로 복원 못해서 대칭키 생성에 문제가 생겼다면 서버는 클라이언트가 무엇을 보내고 있는지 모르므로 적절한 응답을 할 수 없게 되고, 이로써 가짜 서버라는 것이 확인됩니다.

☑ 중간 인증기관 (ICA, Intermediate Certificate Authority)

윈도우즈 운영체제를 사용한다면 [그림 5]와 같이 인증서 관리자 프로그램을 실행시켜 붉은색 사각형으로 표시한 인증기관 분류에 따른 PC에 설치된 인증서들의 목록을 조회할 수 있습니다. 여기서 '신뢰할 수 있는 루트 인증기관'은 [표 1]의 기관들로서 모든 인증서의 최상위에 위치하게 됩니다.

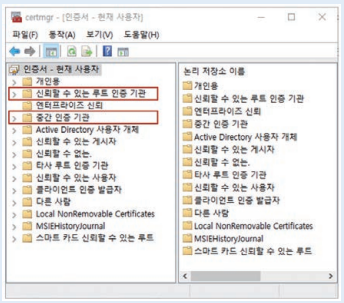


그림 5. PC에 설치된 인증서 종류 및 리스트

그리고 '중간 인증기관'이라는 그룹이 보이는데 중간 인증기관을 구축하면 전반적으로 보안을 강화하면서 인증서 발급 및 관리를 효율적으로 수행할 수 있습니다. 우선 중간 인증기관에 의해 어떤 인증 체계가 형성되는지부터 살펴보겠습니다.

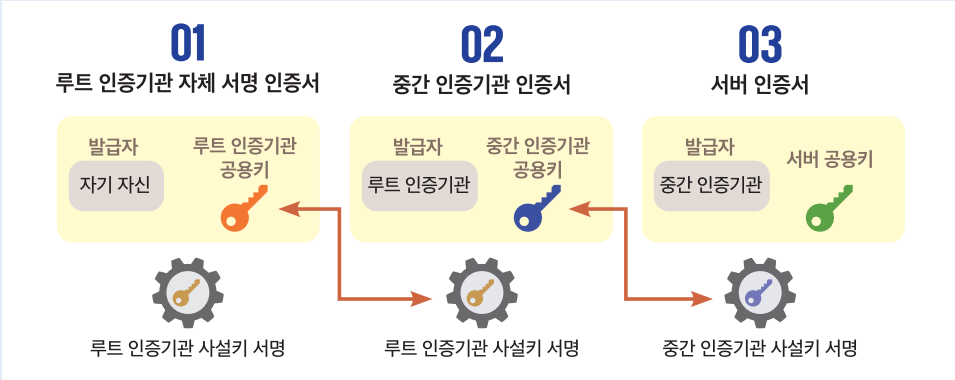


그림 6. 중간 인증기관 인증서

중간 인증기관을 통해 서버에 인증서를 발급하면 [그림 6]의 ③과 같이 중간 인증기관의 사설키로 서명된 인증서를 서버가 갖게 됩니다. 이 인증서의 서명이 중간 인증기관의 공용키로 확인이 되면 이 인증서는 중간 인증기관이 발급한 것이 맞습니다. 그리고 중간 인증기관도 자신의 인증서를 가지는데 이 인증서는 루트 인증기관의 사설키로 서명이 되어 있습니다. 이 인증서 서명이 루트 인증기관의 공용키로 확인이 되면 이 중간 인증기관의 인증서는 루트 인증기관이 발급한 것이 맞습니다.

종합하면 루트 인증기관이 신뢰할 수 있음을 확인해준 중간 인증기관으로부터 받은 서버의 인증서는 결국 루트 인증기관이 신뢰할 수 있다고 확인해준 것과 같은 일종의 신뢰 사슬이 형성되는 것입니다. 이렇게 중간 인증기관을 설치하면 중간 인증기관의 관련 조직이나 부서 등에 인증서 발급, 갱신 및 관리 등의 업무를 효율적으로 운영할 수 있으며 루트 인증기관의 하위에 중간 인증기관을 두어 중간 인증기관이 인증서 발행을 담당하게 함으로써 중간 인증기관의 사설키 유출 등의 사고 발생 시 해당 중간 인증기관의 인증서 폐기(Certificate Revocation) 등을 통해 피해를 줄이고 효과적으로 대처할 수 있습니다.

지금까지 HTTPS를 위한 SSL/TSL 인증서에 관한 내용을 전해드렸습니다. 다음 편에서는 SSL/TSL 핸드셰이킹 프로세스에 관한 내용을 다루겠습니다.

P.S.

C군이 여러분께 전하는 내용 중 전문적 성격이 짙은 것은 엄밀한 언어를 사용하여 설명하기에는 한계가 있습니다.
본 내용은 설명하는 대상에 대한 전체적 맥락의 이해에만 이용하시고, 그 이상은 권위 있는 전문자료를 참고하시기 바랍니다.