about PMI

by 조현래 hrcho@pentasecurity.com

이 문서는 근래에 새로이 표준화된 PMI 와 관련한 몇 가지 용어와 개념에 관해 간략히 정리하고 설명할 것을 목적으로 한다.

PKI

PKI 는 강력한 보안 기반으로써 기능하지만 시스템 통합의 입장에선 그리 큰 매력을 가지지 못한다. 공개키 확인서라는 것은 소유자만을 전제한 것이고, 모든 기능이 소유자의 식별(인증)에서부터 시작한다. 하지만 실제의 시스템은 소유자의 식별 기능과 함께 소유의 대상이 되는 것에 대한 접근 통제 정책을 필요로 한다. 아래에 설명할 AC(Attribute Certificate)는 접근 통제 정책을 보관하기에 알맞은 것으로써 PKI 기반의 시스템 통합에 큰 활용이 예상된다.

개선된 X.509 4th Edition (2000)은 주요소로서 PKI, PMI, Directory Schema 를 포함한다. 따라서 과거에 CA 중심으로 구성되던 시스템은 CA/RA, AA, User Object, Target Object, PKC(공개키 확인서), AC 등으로 구성될 것이다. 이것은 앞으로 X.509 가 인증과 허가 및 Directory-Enabled Application 의 기반으로 작용할 것을 의미한다.

Attribute Certificate

생소할 지 모르겠지만 Attribute Certificate 은 이미 수 년 전부터 정의되고 논의되었던 것이다. 이것의 일차적 용도는 공개키 확인서의 항목으로 넣기에는 부적당한 정보를(관리의 주체가 CA 의 관리자와 다르거나 가변적이어서) 따로 떼어내어 정의함으로써 빈번한 CRL 의 발급을 막고 관리의 주체를 명확히 구분하기 위한 것이다. 공개키 확인서의 취소는 상위 CA 간의 연쇄적인 절차를 거쳐 복잡하게 수행된다. 만약 가변적인 정보가 이런 공개키 확인서에 기록된다면 CRL 관리의 부담으로 인해 공개키 기반 구조의 장점을 살리기 힘든 상황에 이를 것이다.

AC 는 위의 상황을 막기위해 고안된 것으로, 제공하고자 하는 정보(Privilege)와 binding 된 공개키 확인서에 대한 정보(즉 소유자의 공개키는 직접 가지고 있지 않으며 공개키 확인서와 一 대 多로 대응될 수 있음을 의미한다.) 및 AA(Attribute Authority)의 서명을 포함하고 있으며 갱신 방법은 PKC 와 마찬가지로 ACRL 을 관리할 수도 있으나, 짧은 기간의 life time 을 가지며 revocation 하지 않는 정책을 쓸 수도 있다. 즉 주기적으로 갱신되는 것을 전제로 하여 CRL 의 부담을 피할 수도 있다.

Directory-Enabled Application

분산 환경을 통합하는 데는 여러가지 방법이 있고 실제 많은 연구가 진행되고 있으며 성과 또한 많다. 하지만 Common Directory 의 구축은 손쉬운 또 하나의 방법을 제시한다.

잘 정의되지 않은 네트웍 환경에서는 저마다 주소록을 가지고 있고 저마다 개인의 e-mail 을 관리한다. 이것은 관리자에겐 큰 부담이다. 만약 이런 반복 관리되는 정보가 중앙에서 일원화되어 관리되고 어플리케이션이 자신의 설정파일을 따로 관리하지 않고 중앙의 정보를 이용한다면 훌륭하게 분산환경을 통합할 수 있을 것이다. 이를 위해서 Directory Server 는 합리적인 선택이다.

이러한 시스템 구축에서 승패의 관건은 합리적인 스키마(schema)를 구축하여 각각의 어플리케이션이 혼란에 빠지지 않고 통일된 방법으로 정의된 Object 에 접근할 수 있도록 하는 것이다. 즉 반복 언급되는 자료형의 표준을 잘 정의하는 것이다. 현재 이런 작업이 계속 진행되고 있으며 앞으로는 온전한 Common Directory 를 기대할 수 있을 것이다.

PMI

PMI 는 기존의 PKI 가 제공하지 못하는 시스템 통합의 필수 요소인 권한 체계를 위한 기반이다. 이것은 또한 디렉토리 서버의 '충분한' 활용을 의미하기도 한다. PKI 에서 인증서의 발급과 CRL 의 갱신이 핵심이라면 PMI 에서는 AC 의 발급과 이와 관련한 객체의 정의와 관리가 핵심이다.

이것은 간단하지만 아주 강력한 시스템 통합의 도구이다. Directory 에 정의될 수 있는 객체는 단순히 사용자와 조직 뿐 만 아니라 반복 언급되고 객체화 할 수 있는 모든 자료형을 대상으로 한다. 이것은 Directory-Enabled Application 의 기반이 된다.

Push & Pull

발급된 AC 의 활용은 크게 Push 및 Pull 모델로 운용할 수 있다. 즉 서비스를 제공하는 서버가 클라이언트에 대한 인증을 마치고 서비스 허가 여부를 결정하기 위해 직접 디렉토리 서버에서 해당 클라이언트 실행자의 AC 를 가져오는(Pull) 방식과 클라이언트가 서버에 서비스를 요구할 때 AC 를 첨부하는(Push) 방식이다. legacy system 을 통합할 때 Pull 방식은 서버만 수정하면 된다는 장점에 반해 네트웍 트래픽을 많이 사용한다는 단점을 가지고 있다.

AA

접근 통제 체계를 달리하는 조직마다 AA 를 하나씩 두어서 자신의 관리 대상만 관리하는 방식을 취할 수 있다. 사용자는 자신이 사용하고자 하는 시스템 체계를 선별해서 로그인하는 것이 가능하다는 뜻이다. 이것은 RBAC 의 활용 및 다양한 접근 통제 방법론을 적용하기 위한 기반이 된다.

AA 또한 hierarchy 를 가질 수 있으나 CA 의 경우와는 달리 권한의 위임(delegation of privilege)에 관련한 것이다.