通信情報保護學會誌
第 10 券 第 4 號, 2000. 12

속성인증기술과 PMI

진 승 헌*, 최 대 선*, 조 영 섭*, 윤 이 중**

요 약

공개키기반구조(Public Key Infrastructure, PKI)는 인터넷 전자상거래를 위한 정보보호 기반구조로써 많이 활용되고 있다. 그러나 공개키 인증서는 사용자의 신원확인 정보만을 제공하기 때문에 사용자의 신원확인 정보뿐만 아니라 사용자의 권한 정보가 필요한 일반 응용 환경에서는 그 사용이 제한적일 수밖에 없다. 따라서 최근에는 사용자의 권한, 지위, 임무 등과 같은 사용자의 속성 정보를 제공하여 공개키 인증서의 제한적인 기능을 확장하려는 연구가 진행되고 있다. 본 고에서는 인증서 구조에 사용자에 대한 속성 정보를 제공하여 권한 관리가 가능하도록 하는 속성 인증서 기술과 속성 인증서를 발급, 저장, 유통을 제어하는 기반구조인 권한관리기반구조 (Privilege Management Infrastructure, PMI)에 대하여 설명한다.

Ⅰ. 서 론

최근 몇 년 사이에 인터넷을 통한 전자상거래가 매우 급격히 신장하고 있다. 인터넷은 네트워크 자체가 공개적인 구조를 가지고 있기 때문에 정보보호에 대한 연구 및 개발이 매우 활발히 진행되고 있다. 특히 공개키에 대한 인증서를 이용하여 정보보호 메커니즘을 제공하는 기반 구조인 PKI는 전자상거래 보안을 위한 기반 구조로 많은 연구 및 제품 개발이 진행되고 있다. 현재 PKI는 1998년 1.2억 $에서 2003년에는 13억$ 정도로 시장이 확장될 것으로 예측하고 있을 정도로 그 시장 규모가 급격히 팽창하고 있으며 향후 전자상거래 정보보호 메커니즘의 핵심이 될 것으로 전망되고 있다. 그러나 PKI에서 사용하는 공개키 인증서는 사용자에 대한 신원(Identity) 확인 기능만을 제공하며 사용자의 권한, 임무, 지위 등과 같은 사용자에 대한 다양한 속성(Attribute) 정보를 제공하지 못하고 있다. 실제 많은 응용 분야에서는 단순한 신원확인 정보뿐만 아니라 사용자의 권한(Privilege) 또는 속성에 대한 정보를 필요로 하고 있기 때문에 공개키 인증서의 이러한 특징은 PKI 활용의 폭을 제한하고 있다. 따라서 최근에는 사용자의 권한, 지위, 임무 등과 같은 사용자의 속성 정보를 제공하여 공개키 인증서의 제한적인 기능을 확장하려는 연구가 진행되고 있다.

IETF, ITU-T 등과 같은 국제 표준화 단체에서는 속성 정보를 관리 기술에 대한 표준화 작업을 진행 중에 있다. 또한 외국의 몇몇 업체에서는 자체 PKI 제품군에 속성 인증 기술을 지원할 수 있도록 제품의 기능을 개선하고 있는 실정이다. 반면 국내에서는 아직 속성 인증 기술에 대한 연구가 미약한 실정이다.

본 고에서는 사용자의 속성 정보를 제공할 수 있는 방법으로 최근 연구되고 있는 속성 인증서(Attribute Certificate) 기술에 대하여 설명하고 속성 인증서의 발급, 저장 및 유통을 원활히 수행하도록 하는 기반구조인 PMI(Privilege Management Infrastructure)에 대하여 기술한다. 본 고의 구성은 다음과 같다.

2장에서는 속성 인증서의 출현 배경에 대하여 설명하고 3장에서는 속성 인증서와 관련된 다양한 기술적 이슈에 대하여 설명한다. 4장에서는 속성인증서의 프로화일을 살펴보고 5장에서는 속성 인증서의 검증 방법에 대하여 기술한다. 6장에서는 PMI의 모델에 대하여 기술하고 마지막으로 7장에서 결론을 맺는다.

* 한국전자통신연구원 인증기반연구팀 (ETRI, Certification Infrastructure Research Team)
** 정보보호시스템연구부 (Information Security System Department)

2. 속성 인증서

X.509 공개키 인증서는 사용자의 신원(Identity)과 사용자의 공개키를 바인딩함으써 사용자의 신원확인 정보를 위조할 수 없도록 하는 기능을 제공하는 구조로 구성되어 있다. 그러나 실제 응용 서비스 환경에서는 서비스에 접근하려는 사용자들에 대한 선지식이 없는 경우에 단순한 신원 정보 이상을 요구하는 것이 일반적이다. 예를 들어 사용자가 어떤 물품을 주문했을 때, 서비스 시스템에서 물품을 배달할 것인지 아닌지에 대한 결정은 단순한 사용자의 신원 정보를 통해 이루어지지 않는다. 즉 서비스 시스템의 물품 배달 결정에는 사용자가 주문한 물품 대금을 지불할 능력이 있는지를 확인하는 것이 더 중요하다. 이때는 신용카드회사의 사용자에 대한 신용 정보 또는 은행에서의 사용자 신용 정보가 중요하게 된다. 이와 같이 일반적인 응용 서비스 환경에서는 사용자의 신원 정보보다는 오히려 사용자의 권한, 임무, 지위 등과 같은 다양한 사용자의 속성 정보가 더 필요하다. 따라서 최근에는 사용자의 속성 정보를 응용 환경에서 원활히 전달할 수 있는 방법에 대한 연구가 많이 이루어지고 있다.

사용자의 속성 정보를 제공하기 위한 방법으로는 우선 기존의 X.509 공개키 인증서 구조를 활용하는 방법을 사용할 수 있다. 즉 X.509 인증서의 강력한 기능인 확장 필드를 이용하여 사용자의 다양한 권한을 제공할 수 있다. 이 방식은 기존의 공개키 구조를 그대로 이용하면서 속성 정보를 제공할 수 있기 때문에 시스템의 큰 변경 없이 속성 정보를 제공할 수 있다는 장점을 가지고 있다. 그러나 속성 정보는 다음과 같은 두 가지 특징을 가지고 있어 이 방식은 여러 가지 단점을 가지게 된다.

첫번째로 사용자의 속성 정보의 유효기간은 기존 공개키 인증서 내용의 유효기간과 다르다는 특징을 가진다. 즉 공개키 인증서의 경우에는 대부분 유효기간이 1년 또는 그 이상인 것이 일반적이다. 그러나 사용자의 속성 정보는 상대적으로 그 변경이 자주 발생한다는 특징을 가지고 있다. 따라서 공개키 인증서에 사용자의 속성 정보를 저장할 경우에는 속성 정보의 변경으로 인하여 공개키 인증서의 생명주기(life cycle)가 짧아지는 문제가 발생한다. 이것은 공개키 인증서가 상대적으로 자주 폐기되어 인증서의 재발급 비용이 늘어나고 인증서폐기목록 관리 비용이 증가하는 문제를 발생시킨다.

두번째로 공개키 인증서 발급자와 속성 정보를 사용자에게 할당하는 발급자가 서로 일치하지 않다는 특징을 가진다. 인증기관은 공개키 인증서의 사용자 관련 정보를 직접 또는 간접적으로 생성하고 공개키 인증서를 발급한다. 그러나 사용자의 지위, 임무, 은행 신용도 등과 같은 속성 정보는 인증기관에서 결정하는 정보가 아니다. 즉 사용자의 지위와 같은 속성은 사용자가 소속된 회사에서 그 속성을 결정하고 사용자의 신용등급과 같은 속성은 은행 또는 신용평가 기관이 속성 정보를 결정하는 주체가 된다. 따라서 이와 같은 속성 정보를 결정하여 발급하는 기능은 사용자의 속성 정보를 생성하고 할당하는 기관에서 처리되는 것이 합리적이다.

공개키 인증서에 사용자의 속성 정보를 포함할 때 발생하는 이와 같은 문제를 해결하기 위해서 최근에는 사용자의 속성 정보를 저장하는 속성 인증서에 발급하여 사용자의 신원확인을 위한 공개키 인증서와 분리하여 관리하는 방식이 연구되고 있다. 이 방식은 사용자의 공개키와 신원 정보는 기존의 공개키 인증서를 그대로 활용하며 사용자에 대한 속성 정보는 속성 인증서로 발급한다. 사용자의 속성 인증서는 사용자에 대한 공개키 인증서를 가리키도록 되어 있어 사용자의 공개키 인증서와 바인딩이 된다. 이 방식은 인증서 발급 기능이 적절하게 분리되며 명확해진다. 이것은 중앙의 인증기관에서는 사용자에 대한 공개키 인증서를 발급하고 상대적으로 공개키 인증서 소유주에 대한 속성을 직접적으로 관리하는 곳에서 속성 인증서를 발급하는 방식을 이용하게 된다. 이 경우 각각의 사용자는 하나의 공개키 인증서에 대하여 각기 다른 단체에서 발급한 여러 개의 속성 인증서를 소유할 수 있다. 예를 들어 법률가는 자신이 속한 법률 회사에서 발급한 속성 인증서뿐만 아니라 법률 협회에서 발급한 속성 인증서 그리고 자신이 접근하는 서비스에 따라 선택적으로 다양한 속성 인증서를 가질 수 있다.

3. 속성 인증서 관련 이슈

본 장에서는 속성 인증서의 도입에 따라 발생하는 다양한 기술적인 이슈에 대하여 살펴본다.

3.1 속성인증서 분배 방식

속성 인증서를 분배하는 방식은 pull 모델과 push

모델 두 가지가 사용된다. pull 모델은 속성 인증서가 생성되었을 때 디렉토리에 속성 인증서를 게시하는 방식이다. 따라서 속성 인증서를 사용하는 응용 서비스는 속성 인증서가 필요할 때 디렉토리에서 인증서를 검색하여 사용한다. 반면 push 모델은 사용자가 응용 서비스에 접근할 때 속성 인증서를 직접 전달하는 방식이다. 이 방식은 사용자가 응용 서비스에 접근할 때 사용자 이름과 패스워드를 전달하는 것과 같은 방식이다. Pull 모델은 클라이언트 또는 클라이언트-서버 프로토콜의 변경 없이 구현될 수 있다는 장점을 가지고 있으며 클라이언트의 권한이 서버 도메인 내에서 할당되어야만 하는 경우에 특히 적합한 모델이다. push 모델은 기반구조를 단순화시키고 디렉토리를 필요로 하지 않는다. 또한 디렉토리 서버에서 속성 인증서 검색을 위한 서버의 추가적인 통신비용이 필요하지 않기 때문에 시스템 성능이 개선될 수 있다는 장점을 가지고 있다. 이 두 모델은 응용 서비스 환경의 구성과 요구 사항에 따라 선택적으로 활용될 수 있다. 다음 [그림 1]은 속성 인증서의 교환 방식을 보인다.

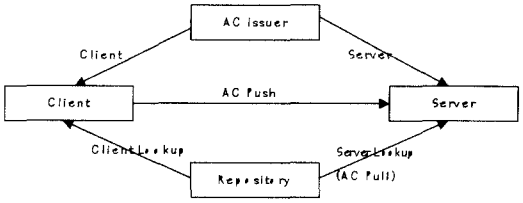


그림 1. 속성 인증서 교환 방식

[그림 1]에서 AC Issuer는 속성 인증서 발급 기관을 의미하며 Repository는 속성 인증서를 게시하는 디렉토리를 나타낸다. Repository에 저장된 속성 인증서는 속성인증서인 AC의 발급자에 의해 저장된다. Server가 Repository에서 속성 인증서를 검색하는 방식이 Pull 방식이며 Client에서 Server에 접속할 때 직접 속성 인증서를 전달하는 방식이 Push 방식이다.

3.2 속성 인증서 생명 주기

속성 인증서는 일반적으로 공개키 인증서에 비해 짧은 생명 주기를 가진다. 속성 인증서의 짧은 생명 주기는 기존 공개키 인증서에 대한 인증서폐기목록 관리 기술과는 다른 인증서폐기 관련 기술을 필요로 하게 된다. 먼저 인증서 발급에 관련된 프로세스는 가벼운 프로세스(light process)여야 한다. 이것은 속성 인증서는 그 내용이 자주 변경되기 때문에 인증서 발급이 공개키 인증서에 비해 상대적으로 매우 빈번하게 발생되기 때문이다. 즉, 인증서 발급 작업이 단순하여야 되며 자동적으로 수행될 수 있어야 한다. 또한 속성 인증서의 속성을 직접 생성하고 관리하는 곳에서 속성 인증서에 대한 발급 작업이 이루어져야 한다. 즉 속성 인증서 소유주의 직무와 같은 속성은 실제 다양한 환경에서 의미를 가지며 사용되지만 소유주의 직무가 실제로 변화하는 곳에서 직접 속성 인증서를 발급하는 것이 가장 합리적이다.

마지막으로 속성 인증서에 대한 폐기목록 관리가 필요 없게 될 수 있다. 즉 속성 인증서가 폐기된 상태로 노출되는 기간이 아주 짧으면 폐기목록이 아닌 다른 방법으로 관리가 가능하다. 예를 들어 회사 직원의 속성 인증서가 수 시간의 유효기간만을 가질 경우, 만약 그 직원이 해고된다면 단지 몇 시간 동안만 그 직원을 격리시키는 방법을 사용하여 유효하지 않은 속성 인증서의 사용을 방지할 수 있다.

3.3 속성 인증서 폐기 방식

속성 인증서의 속성이 상당히 오랜 기간 동안 유지되는 서비스 환경에서는 폐기목록 관리가 필요하다. 이와 같은 경우에서는 속성 인증서 폐기 목록(Attribute Certificate Revocation Lists, ACRLs)를 관리할 수 있다. ACRLs는 CRL 관리와 동일한 방법으로 수행된다. 그러나 속성 인증서의 생명 주기는 일반적으로 매우 짧기 때문에 인증서 폐기목록이 필요하지 않는 경우가 대부분이다. 따라서 일반적으로 속성 인증서는 폐기하지 않고 단지 만료(expire) 시키면 된다.

4. 속성 인증서 프로화일

속성 인증서에 대한 기본 구조는 ITU-T X.509 Version 4에서 규정하고 있다. ITU-T X.509 Version 4는 현재 드래프트 상태로 개발 중에 있다. 그러나 ITI-T의 표준은 인터넷뿐만 아니라 다양한 네트워크 환경을 고려한 구조이기 때문에 그 내용이 인터넷에서 활용하기에는 매우 복잡한 실정이다. IETF는 속성 인증 기술을 인터넷 환경에 적합하도록 속성 인증서 프로화일을 개발하고 있다.

본 장에서는 현재 IETF에서 개발중인 속성 인증서 프로화일에 대하여 기술한다.

4.1 표준 필드

본 절에서는 IETF의 속성 인증서 프로화일에서 표준 확장 필드에 대하여 간략히 기술한다. 속성 인증서의 문법은 다음 [그림 2]과 같다.

```
AttributeCertificate ::= SEQUENCE {
  acinfo               AttributeCertificateInfo,
  signatureAlgorithm   AlgorithmIdentifier,
  signatureValue       BIT STRING
}

AttributeCertificateInfo ::= SEQUENCE {
  version              AttCertVersion DEFAULT v1,
  holder               Holder,
  issuer               AttCertIssuer,
  signature            AlgorithmIdentifier,
  serialNumber         CertificateSerialNumber,
  attrCertValidityPeriod AttCertValidityPeriod,
  attributes           SEQUENCE OF Attribute,
  issuerUniqueID       UniqueIdentifier OPTIONAL,
  extensions           Extensions OPTIONAL
}

  AttCertVersion ::= INTEGER { v1(0), v2(1) }
```

그림 2. IETF 속성 인증서 프로화일

■ **Version**

이 필드는 속성 인증서의 버전을 나타낸다.

■ **Holder**

이 필드는 속성 인증서의 소유주를 나타낸다. 이 필드의 값은 기본 공개키 인증서의 ID인 base CertificateID, 인증서 소유주 또는 권한의 이름인 entityName 또는 objectDigestInfo 값으로 설정될 수 있다. 기본 공개키 인증서는 속성 인증서 소유주에 대응되는 공개키 인증서에 대한 식별 값을 나타내며 공개키 인증서의 발급자와 일련번호로 구성된다.

■ **Issuer**

이 필드는 속성 인증서의 발급자를 의미한다. 일반적으로 발급자의 DN 이름으로 구성되어 있다.

■ **Signature**

속성 인증서의 서명 값을 검증하는데 사용되는 알고리즘 식별자를 포함하는 필드이다.

■ **Serial Number**

이 필드는 속성 인증서의 일련번호를 포함한다. 속성 인증서마다 유일하게 식별될 수 있는 정수 값으로 구성된다.

■ **Validity Period**

이 필드는 속성 인증서의 유효기간을 나타낸다. 즉 속성 인증서의 소유주와 속성 필드들의 바인딩이 유효한 기간을 보증하는 필드이다.

■ **Attributes**

이 필드는 속성 인증서의 소유주에 대한 속성 정보를 제공하는 필드이다. 자세한 설명은 4.2절에서 기술한다.

■ **Issuer Unique Identifier**

이 필드는 속성 인증서 발급자를 식별하는 용도로 사용된다. 이 필드는 속성 인증서의 소유주의 공개키 인증서에서 사용되지 않았다면 사용되지 않아야 한다.

■ **Extensions**

이 필드는 속성 인증서 자체에 대한 다양한 정보를 제공하는 필드이다. Attribute 필드가 속성 인증서 소유주에 대한 속성 정보를 제공하는데 비해 이 필드는 속성 인증서 자체에 대한 정보를 제공한다.

4.2 속성 필드

본 절에서는 4.1 절에서 간략히 설명한 Attribute 필드를 자세히 기술한다. Attribute 필드에 대한 문법은 다음 [그림 3]과 같다.

```
attributes          SEQUENCE OF Attribute,

Attribute ::= SEQUENCE {
    type      AttributeType,
    values    SET OF AttributeValue
      -- at least one value is required

}
```

AttributeType ::= OBJECT IDENTIFIER
AttributeValue ::= ANY DEFINED BY AttributeType

그림 3. IETF 속성 인증서의 속성 필드

속성 필드는 각각의 속성으로 구성된 집합이다. 또한 각각의 속성은 속성 타입과 속성 값의 집합으로 구성된다. 속성 타입은 각각의 속성을 유일하게 식별할 수 있도록 유일한 Object ID를 가지며 속성 필드에 하나만 존재할 수 있다. 반면에 속성은 하나 또는 여러 개의 속성 값을 가질 수 있다.

속성 인증서의 속성 필드에 포함될 수 있는 속성은 다음과 같다.

■ Service Authentication Information

이 필드는 서비스 이름, 소유주 식별정보, 선택적인 인증 정보로 구성되며 응용 서비스가 사용자를 인증할 때 사용되는 필드이다. 일반적으로 응용 서비스 환경에서 사용자 이름과 패스워드를 식별하는데 사용된다.

■ Access Identity

이 필드는 사용자의 접근 권한을 기술하는 필드이다. 즉 서비스 시스템이 속성 인증서 소유주의 접근 권한을 검증할 수 있도록 접근 관한에 대한 정보를 기술하며 여러 가지 값을 가질 수 있다. 그러나 패스워드와 같은 사용자에 대한 인증 정보는 포함되지 않는다.

■ Charging Identity

이 필드는 서비스 시스템이 속성 인증서 소유주에게 과금(Charging)을 할 수 있도록 고안된 필드이다.

■ Group

이 필드는 속성 인증서 소유주의 그룹 멤버쉽에 대한 정보를 제공하는 필드이다.

■ Role

이 필드는 속성 인증서의 소유주에 대한 임무(Role)를 표현하는 필드이다. 예를 들면 소유주가 시스템 관리자, 프로그래머 또는 일반 사용자인지를 기술하는데 사용된다.

■ Clearance

이 필드는 속성 인증서 소유주의 보안 등급에 대한 정보를 포함한다.

4.3 확장 필드

속성 인증서의 확장 필드에는 다음과 같은 확장이 포함될 수 있다.

■ Audit Identity

이 필드는 감사기록으로 속성 인증서 소유주 이름 필드를 직접 사용하는 것이 부적합한 서비스 환경에서 감사기록을 위해 사용된다. 즉, 서비스 시스템이 속성 인증서 소유자의 시스템 사용에 대한 감사/기록을 수행해야 되지만 속성 인증서 소유자의 신원 정보를 직접 접근해서는 안되는 경우에 사용되는 필드이다.

■ AC Targeting

이 필드는 속성 인증서 소유자가 접근할 수 있는 목적(Target) 서비스를 기술하는데 사용되는 필드이다. 이 필드에 속한 서비스 이름에 포함되지 않는 서비스 시스템에서는 속성 인증서 소유주에게 서비스를 제공해서는 안된다.

■ Authority Key Identifier

이 필드는 속성 인증서 검증 모듈이 속성 인증서를 검증할 때 속성 인증서의 서명을 생성한 발급자 인증서를 식별하여 검색하는데 사용되는 필드이다.

■ Authority Information Access

이 필드는 속성 인증서 검증 모듈이 속성 인증서의 폐기 상태를 검증할 때 사용되는 필드이다. OCSP를 이용한 인증서 상태 검색이 가능하도록 OCSP 응답자의 위치를 URI로 기술한다.

■ CRL Distribution Points

이 필드는 속성 인증서의 폐기 여부를 인증서폐기 목록의 분배점을 이용하여 검사할 때 사용되는 필드이다.

■ No Revocation Available

이 필드는 속성 인증서 발급자가 별도의 인증서 상태 검증을 위한 정보를 제공하지 않는다는 것을 의미한다.

5. Attribute Certificate 검증

본 장에서는 속성 인증서의 검증 방법에 대하여 기술한다. 본 장에서 설명하는 속성 인증서의 검증 방법은 IETF의 속성 인증서 프로화일의 검증 방법이다.

속성 인증서 검증 모듈은 기본적으로 속성 인증서의 다음과 같은 항목을 검사한다. 만약 하나라도 검증이 실패하며 속성 인증서의 검증은 실패한 것이다.

1. 루트 인증서부터 속성 인증서를 발급한 발급자 인증서까지 인증경로를 구성한다. 루트 인증서에서 발급자 인증서까지의 인증경로 검증은 RFC 2459의 인증경로 검증 방법을 따른다.
2. 속성 인증서의 서명 값이 올바른지 발급자 인증서의 공개키로 검증한다.
3. 속성 인증서를 생성한 발급자의 공개키 인증서가 올바른지 확인한다. 즉 발급자의 공개키 인증서의 keyUsage 확장 필드가 전자 서명을 검증하는데 유효한지 검사한다. 또한 일련번호와 폐기 방식 등의 혼란을 피하기 위해 속성 인증서 발급자와 일반 공개키 인증서 발급자가 동일해서는 안된다. 따라서 속성 인증서 발급자의 공개키 인증서에서는 basicConstraints 확장의 cA BOOLEAN 필드 값이 TRUE로 설정되어 있으면 안된다.
4. 속성 인증서 발급자가 응용 서비스 환경 설정이나 여타 다른 방법으로 속성 인증서 발급자로 신뢰되어 있는지 확인한다.
5. 속성 인증서를 검증하는 시각이 속성 인증서의 유효기간 내에 있어야 한다. 만약 검증하는 시각이 유효기간의 시작 또는 끝 시각과 같다면 속성 인증서는 적절하다고 판단한다.
6. 검증하는 서비스가 속성 인증서의 Targets 필드와 부합되는지 확인한다.
7. 속성 인증서에 검증 모듈이 지원하지 않는 critical 확장이 있으면 속성 인증서의 검증은 실패한다.

위와 같은 기본적인 속성 인증서 검증이 수행된 후 다음과 같은 추가적인 검증을 수행할 수 있다.

1. 속성 인증서는 속성 인증서 검증 모듈의 환경 설정에 따라 검증이 실패할 수 있다. 예를 들어 검증 모듈이 특정한 속성을 포함하도록 또는 포함하지 않도록 설정되어 있다면 이와 같은 특정 환경 설정에 따라 검증이 실패하거나 또는 성공할 수 있다.
2. 만약 검증 모듈이 응용 서비스에게 속성 인증서의 내용을 질의할 수 있는 인터페이스를 제공한다면 검증 모듈은 환경 설정 정보에 따라 속성 인증서에서 추출한 속성들을 필터링할 수 있다. 예를 들어 검증 모듈은 특정 서버에게 속성 인증서의 특정 속성이 제공되지 않도록 설정될 수 있다.

6. PMI 모델

속성 인증서가 정보보호 메커니즘으로 활용되기 위해서는 속성 인증서의 발급, 저장, 유통이 속성 인증서 생성 기관, 속성 인증서 소유주, 응용 서비스 시스템 등에서 원활이 동작할 수 있어야 한다. PMI는 속성 인증서의 발급, 저장, 유통, 검증 등을 포괄하는 권한관리 기반구조이다. 여기에서 권한은 속성과 같은 의미이다. 따라서 PMI를 구성하는 방법에 따라 속성 인증서의 활용 방식과 응용 서비스 환경이 영향을 받게 된다. 본 장에서는 PMI의 모델에 대하여 기술한다.

6.1 Basic Model

일반적으로 권한 관리 모델은 객체(object), 권한 주장자(privilege asserter) 및 권한 검증자(privilege verifier)의 세가지 개체(entity)로 구성된다.

객체는 응용 서비스 환경에서 보호되는 자원(resource)을 의미한다. 예를 들면 내부 네트워크 시스템을 보호하는 방화벽 또는 파일 시스템의 파일이 객체가 될 수 있다. 이러한 객체는 메쏘드를 가질 수 있다. 즉 방화벽 객체는 Allow Entry 메쏘드를 가질 수 있으며 파일 시스템의 파일은 읽기, 쓰기 그리고 실행 메쏘드를 가질 수 있다.

권한 주장자는 일정한 권한을 소유하며 특정 응용 환경에서 자신의 권한을 주장하는 개체이다. 권한 주장자는 속성 인증서의 소유자와 일치한다. 권한 검증자는 주어진 응용 환경에서 권한 주장자가 제시하는 권한이 적절한지 판단하는 개체이다. 즉 속성 인증서의 검증 모듈에 해당된다. PMI 를 구성하는 기본 모델은 다음 [그림 4]와 같다.

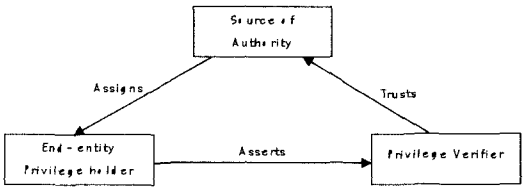


그림 4. 기본 PMI모델

[그림 4]에서 SoA(Source of Authority)는 권한을 부여하는 곳이며 End-entity는 속성 인증서 소유자이다. Privilege Verifier는 서비스 시스템에서 속성 인증서를 검증하는 곳이다. SoA는 End-entity에게 권한을 부여한다. End-entity는 서비스 시스템에 부여 받은 권한을 전달하여 자신이 권한을 소유하고 있음을 주장하는 속성 인증서의 소유주에 해당한다. Privilege Verifier는 End-entity가 소유하고 있다고 주장하는 권한이 자신의 신뢰하는 SoA로부터 부여 받은 권한인지 확인한다.

6.2 Privilege Delegation Model

일반적으로 PMI의 구성은 6.1 절에서 설명한 기본 모델을 사용하지만 일부 특수한 환경에서는 권한을 위임하는 것이 필요하다. 이와 같은 환경에서는 권한 위임 모델(Privilege Delegation Model)을 이용하여 PMI를 구성한다.

권한 위임 모델은 [그림 5]와 같아 구성된다.

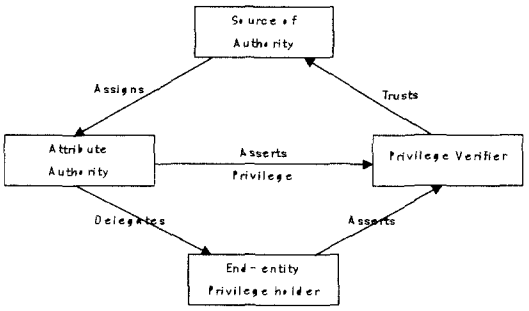


그림 5. 권한 위임 PMI 모델

권한 위임 모델은 SoA(Source of Authority)에서 자신의 권한 할당 기능을 AA(Attribute Authority)에 위임하는 모델이다. Privilege Verifier가 End-entity의 권한을 검증하기 위해서는 자신이 신뢰하는 SoA로부터 AA가 적절한 권한 위임을 받았는지 검증하는 것이 추가적으로 필요하다.

이외에도 PMI모델로는 Control Model, Roles Model 등이 있다.

7. 결론

공개키 인증서는 신원확인 기능만을 제공하기 때문에 공개키 소유자 즉 서비스 시스템 사용자에 대한 다양한 속성 정보를 필요로 하는 응용 서비스에서는 그 사용이 제한적일 수밖에 없었다. 이러한 문제를 해결하기 위해 속성 인증 기술에 대한 연구가 진행되고 있으며 속성 인증서를 이용하여 사용자의 다양한 속성 정보를 처리하게 되었다. 속성 인증서는 기존의 공개키 인증서와는 달리 짧은 생명 주기를 가지고 있다. 이것은 인증서 폐기 관련 기술이 기존 공개키 인증서에서 사용하던 기술과는 다른 기술을 요구하였다. 또한 공개키 인증서 의 발급 기관인 인증기관과는 달리 속성 인증서는 사용자의 속성과 밀접한 관련이 있는 기관에서 발급되는 것이 적합하다는 특징을 가지고 있다. PMI는 이와 같은 속성 인증서가 속성 인증서 발급 기관, 사용자, 응용 서비스에서 원활이 발급, 저장, 전달 및 검증되도록 하는 기반구조이다.

속성 인증서는 신원확인 뿐만 아니라 다양한 속성 정보를 필요로 하는 많은 응용 분야에서 사용될 것이며 이를 위해서는 속성 인증 기술에 대한 많은 연구가 필요하다. 즉 속성 인증서를 short-lived 인증서로 생성할 것인지 또는 공개키 인증서와 같이 상대적으로 긴 유효기간을 가지는 인증서로 생성할 것인지 판단할 수 있는 다양한 환경 요소가 정량화 될 수 있는 연구가 필요하다. 또한 PMI의 구성을 어떻게 하는 것이 효율적인지 기존 PKI와 연동하는 방법에 대한 많은 연구가 필요하다.

참 고 문 헌

[1] ITU-T "Draft Revised ITU-T Recommendation X.509(2000 E) : Information Technology Open Systems Interconnection - The Directory : Public-Key and Attribute Certificate Frameworks"

[2] RFC 2459, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile",

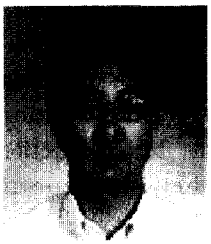
IETF PKIX Working Group, January. 1999
[3] Internet Draft, "An Internet Attribute Certificate Profile for Authorization", IETF PKIX Working Group, August, 2000
[4] J. J. Hwang, K. C. Wu, D. R. Liu, "Access control with role attribute certificate", Computer Standards & Interfaces, Vol.22, pp.43 ? 53, 2000
[5] Entrust, http://www.entrust.com, 2000
[6] Baltimore, http://www.baltimore.co.uk, 2000

〈著者紹介〉

진 승 헌 (Seung-hun Jin)
1993. 2 : 숭실대학교 전자계산학과 학사
1995. 2 : 숭실대학교 전자계산학과 석사
1994. 12 ~ 1996. 4 : 대우통신 종합연구소
1996. 5 ~ 1999. 5 : 삼성전자 통신연구소
1999. 6 ~ 현재 : 한국전자통신연구원 정보보호기술연구본부

최 대 선 (Dae-seon Choi)
1995. 2 : 동국대학교 컴퓨터공학과 학사
1997. 2 : 포항공과대학교 컴퓨터공학과 석사
1997. 1 ~ 1998. 6 : 현대전자 정보시스템 연구소
1998. 7 ~ 1999. 6 : 현대정보기술 연구소
1999. 7 ~ 현재 : 한국전자통신연구원 정보보호기술연구본부

조 영 섭 (Yeong-sub Cho)
1993. 2 : 인하대학교 전자계산공학과 학사
1995. 2 : 인하대학교 전자계산공학과 석사
1999. 2 : 인하대학교 전자계산공학과 박사
1998. 12 ~ 현재 : 한국전자통신연구원 정보보호기술연구본부



윤 이 중 (E-Joong Yoon)
1988. 2 : 인하대학교 전산과 학사
1990. 2 : 인하대학교 전산과 석사
1997. 2 - 현재 : 충남대학교 컴퓨터과학과 박사과정
1990. 2 ~ 1999. 1 : 한국전자통신연구원 부호기술연구부
1999. 1 ~ 2000. 3 : 한국전자통신연구원 정보보호기술연구본부 인증기반팀장
2000. 3 ~ 현재 : 한국전자통신연구원 정보보호기술연구본부 정보보호시스템연구부장