PMI용 인증서 검증 효율성 개선 방법 연구

김상천[0] 송주석
연세대학교 컴퓨터과학과
{sckim, jssong}@emerald.yonsei.ac.kr

Efficient Verification of PMI's Certificate

Sang-Chun Kim[0] Joo-Seok Song
Dept. of Computer Science, Yonsei University

요 약

본 연구에서는 권장 검증 경로와 검증된 공개키 인증서 포리스트를 이용하여 PMI용 인증서 검증 시간과 노력을 줄일 수 있는 방법을 제시한다. 권장 검증 경로는 상위 권한 인증 기관에서 사전에 구성한는 검증 경로 정보로서 PMI용 인증서의 확장영역을 통해 전달된다. 권한 검증자는 이를 이용하여 인증서의 검색/획득 과정과 검증의 일부분을 병렬적으로 처리하며 더 빠른 시간 내에 검증을 완료할 수 있다. 또, PMI용 인증서 검증 시에는 여러 개체의 신원 인증 과정이 필수적인데, 이러한 신원인증과정에서 검증된 공개키 포리스트를 이용하여 기존의 검증 정보를 재사용 할 수 있도록 하였다.

1. 서 론

권한 관리 기반 구조(Privilege Management Infrastructure, PMI)[1,2,3]는 공개키 기반 구조(Public Key Infrastructure, PKI)[1,2]와 연계하여 권한 관련 자원과 권한 소유자간의 관계를 신뢰기간이 보증하고 유지하는 구조를 말한다.

권한관리 기반 구조에서 사용되는 인증서를 권한 관리 기반 구조용 인증서(PMI용 인증서) 또는 속성 인증서라 하며 현재 IETF와 ITU-T와 같은 국제 표준화 단체에서 표준화가 진행중이다. 이러한 속성 인증서 검증 알고리즘은 PKIX의 Draft[3]에 기술되어 있다.

본 연구에서는 이러한 검증 과정을 효율적으로 수행하기 위해서 권장 검증 경로(Recommended Verification Path, RVP)를 속성 인증서의 확장영역에 추가하여 검증과정이 병렬적으고 더 효율적으로 진행될 수 있도록 제안하였으며, 이와 더불어 권한 검증자가 검증된 공개키 인증서 포리스트(Verified Public-key Certificate Forest, VPCF)를 유지하여 검증에 참여하는 개체의 신원 인증과정에서 기존의 검증 정보를 재 사용할 수 있도록 하였다.

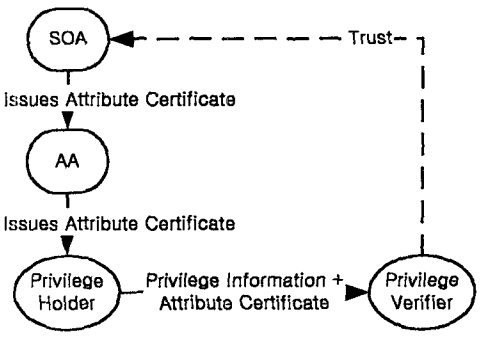
본 논문의 구성은 다음과 같다. 제 2 장에서는 속성 인증서의 검증 기법을 설명하고 제 3 장에서 권장 검증 경로와 검증된 공캐키 인증서 포리스트를 이용한 속성 인증서 검증과 그 효과를 기술하고 분석한다. 그리고 제 4 장에서 끝으로 결론을 맺는다.

2. 관련연구

2.1 권한 관리 기반 구조와 속성 인증서

권한 관리 기반 구조는 최상위 속성 인증기관(Source of Authority, SOA), 속성 인증기관(Attirbute Auththority, AA), 권한 소유자(Privilege Holder), 권한 검증자(Privilege Verifier) 등의 요소로 이루어져 권한 인증 서비스를 제공하게 된다.[1]

권한 관리 기반 구조에서는 상위 속성 인증기관(최상위 속성 인증기관 포함)에서 하위 속성 인증기관 또는 말단 속성 인증기관에서 속성 소유자에게로 속성 인증서를 통해 해당 속성 정보를 인증해 준다. 속성 소유자는 이렇게 발급된 속성 인증서를 속성 검증자에서 제시함으로서 자신의 권한을 주장할 수 있고, 속성 검증자는 속성 인증서를 검증하여 속성 소유자의 권한 정보를 신뢰할 수 있게 된다.



(그림 1) 권한 관리 기반 구조

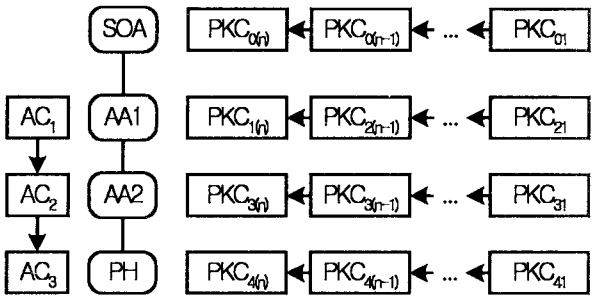
이러한 권한 관리 기반 구조에서 사용되는 속성 인증서는 공개키 인증서와는 달리 소유자의 공개키를 포함하지 않는다. 즉, 공개키 인증서가 신원과 공개키 간의 관계를 보증하다면 속성 인증서는 신원과 해당 속성 간의 관계를 보증하는 역할을 한다. 속성 인증서는 표준 속성(attributes), 확장영역(extensions), 발급자의 서명(signature) 등의 주요 항목으로 구성되어 있다.

2.2 권한관리 기반 구조용 인증서 검증

PKIX의 Draft[3]에 설명되어 있는 속성 인증서의 검증 방법은 공캐키 인증서를 이용한 속성 소유자의 신원 검증, 속성 인증서의 발급자 서명 검증, 발급자의 신원 검증, 유효기간 검증

등의 작업으로 이루어진다.

그러나 이것은 SOA나 AA들 간에 권한 위임이 발생하지 않은 단순한 구조를 가정한 것으로 실제로는 공개키 인증서의 검증 때와 같이 인증서 체인의 각 단계를 위와 같은 방법으로 검증하여 한다. 즉 아래의 (그림 2)에서와 같이 속성 인증서 체인을 따라가며 속성 소유자의 속성 검증해야 하며, 이 때 각 단계에서 참여하는 개체들의 신원 확인을 위해 해당 개체의 공개키 인증서 별로 공개키 인증서 체인 검증도 수반되어야 한다



(그림 2) 속성 인증서 검증 체인

따라서 하나의 속성 인증서를 검증하는 것은 공개키 인증서를 검증하는 것보다 몇 배 이상의 많은 계산량이 필요하게 된다.

(그림 2)와 같은 환경에서 속성 검증자가 속성 인증서의 검증하는 과정을 생각해 보자. 우선 속성 검증자는 속성 소유자(PH)의 속성 인증서(AC_3)를 보고 속성 소유자 공개키 인증서($PKC_{4(n)}$)을 찾아 속성 소유자의 신원을 확인하게 된다. 이 과정에서 속성 소유자의 공개키 인증서 체인 $PKC_{4(n)}$-$PKC_{4(n-1)}$-...-PKC_{41}을 구성/검증하여야 한다. 이후 AC_3의 발급자 필드를 확인하여 속성 인증서 발급자인 말단 속성 인증기관(AA2)의 공개키 인증서를 획득하여 PKC 체인을 구성해야 한다. 이후의 속성 인증서 체인 구성도 같은 방법이 반복 적용되게 된다. 이과정에서 속성 검증 등의 작업도 수행된다.

3. PMI용 인증서의 효율적인 검증 방법 제안

3.1 권장 검증 경로(RVP)

실제로 위의 속성 인증서 검증 과정에서 속성 인증서와 공개키 인증서 경로 구성은 속성 검증자가 매번 수행하지 않아도 되는 부분이다. 공개키 기반 구조에서도 검증 경로를 미리 구성하여 검증의 효율성을 높이기 위해 인증 트리의 유효성을 확인하는 문서(Certification Tree Validity Statement, CTVS)[4]를 발행하는 방법이 연구되기도 하였다. 그러나 말단 CA 등이 CTVS를 발행하고 검증 경로에 대한 정보를 보증하며 이에 대한 법적인 책임을 지는 방법은 하위 CA가 상위 CA의 인증서 상태를 항상 주시하고 있어야 하고 상위 CA의 인증서 상태를 잘못 조회한 점에 대해 불필요한 책임을 지는 등의 유연성이 부족한 문제를 가지고 있다. 더구나 과도한 책임 부과로 인해 말단 CA 등과 같은 CTVS를 발행하는 주체가 CTVS의 발행을 회피할 우려가 있다. 더구나 속성 인증서의 경우 CRL을 사용하지 않는 경우도 있기에 CRL과 비슷한 방법으로 운영되는 CTVS는 속성 인증서 체인 검증에 응용하기에는 문제점이 있다. 또 인증서의 인증 경로를 일련 번호를 이용하여 저장하는 방법[6]도 일련 번호 길이의 제약이 문제가 되며 속성 인증서와 발행이 너무 빈번하기에 적합하지 않다.

여기서는 권장 검증 경로를 이용하여 속성 인증서 및 공개키 인증서의 체인을 재사용성을 높이면서 체인 구성의 유연성을 확보하며 체인을 구성한 주체에 불필요한 책임을 지우지 않는 방법을 제안한다.

권장 검증 경로는 (그림 3)과 같은 non-critical 확장영역(non-critical extension)으로서 말단 속성 인증기관이 속성 인증서를 발행하는 과정에서 미리 구성하여 포함시키게 된다.

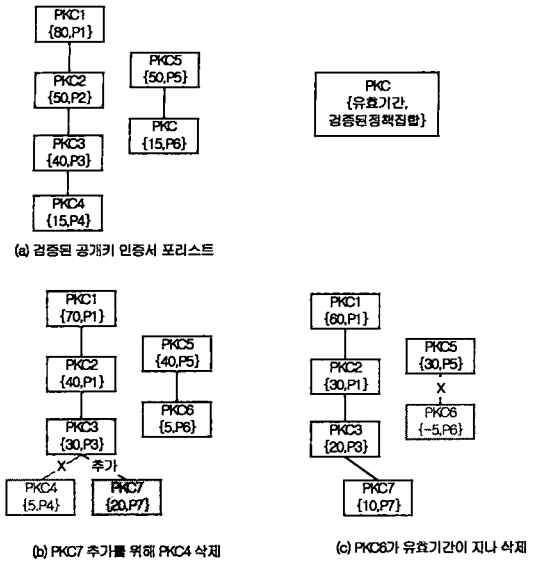
```
name            id-recommendedVerificationPath
OID             { ... }
syntax          RecommendedVerficationPaths
criticality     MUST be FALSE

RecommendedVerificationPaths ::= SEQUENCE {
        pathLength          INTEGER,
        aCInfo              AttributeCertificateInfo
}

AttributeCeritificateInfo ::=SEQUENCE {
        isSOA               BOOLEAN,
        aC                  Holder,
        CHOICE {
            pKC EXPLICIT     [0] AttCertIssuer,
            pKCPath EXPLICIT [1] PKCPath;
        } OPTIONAL
}

PKCPath ::= SEQUENCE {
        pathLength          INTEGER,
        SEQUENCE OF         AttCertIssuer
}
```

(그림 3) 권장 검증 경로(Recommended Verification Path)의 ASN1. 표기

3.2 검증된 공개키 인증서 포리스트(VPCF)

속성 인증서의 검증 과정에서는 많은 개체의 신원 인증이 필수적이다. (그림 2)에서와 같이 PH, AA2, AA1, SOA의 신원 인증 시 매번 공개키 인증서 체인이 구성하는 것은 필수적인 과정이다. 이러한 공개키 인증서 체인은 전혀 일치하지 않는 경우도 있지만 부분적으로 일치하거나 말단의 몇 개의 CA만 다르고 체인의 대부분이 같은 경우도 많다. 속성 검증자는 대개 서비스를 제공해주는 서버 입장인 경우가 많은데 이러한 서버 내의 안전한 장소에 기존의 검증된 인증 경로를 저장하여 다른 개체의 신원 인증에 이용할 경우 더 빠른 속성 인증서 검증이 이루어질 수 있다.

다수의 루트CA에 대해 이러한 경로를 저장하면 결국 (그림 4)와 같은 포리스트가 형성되게 되고 이를 검증된 공개키 인증서 포리스트(VPCF)라 부른다. 이 포리스트의 각 노드에는 인증 경로가 유효한 기간(PKC의 유효기간, CRL 발행 예정일, 상위 노드의 유효기간의 교집합)과 RootCA로부터 해당 노드까지 검증 후의 정책 집합(Policy Set)이 저장되게 된다. 이러한 노드들은 새로운 개체의 신원 인증 시 추가된 PKC의 검증 결과를 바탕으로 추가되며 유효 기간이 지나면 삭제된다. 또한 검증된 정보는 안전한 곳에 저장되어야 하고 이러한 장소는 한정되어있기 마련이므로 저장공간이 차면 노드들은 생성 시간, 최후 접근 시간, 접근 빈도, 유효 기간, 트리 내의 위치 등의 요소 중 일부 또는 전체가 고려되어 삭제된다.



(그림 4) 검증된 공개키 인증서 포리스트의
노드 추가와 삭제

3.3 분석

권장 검증 경로를 이용하면 검증에 필요한 인증서를 병렬적으로 검색/획득할 수 있으며 부분적으로 병렬적인 검증 또한 가능하다. 즉 인증서 체인의 길이에 상관없이 검증에 필요한 인증서들을 동시에 검색/획득하여 검증에 사용할 수 있는 것이다. 또 상호 인증 등으로 인해 인증 경로가 다양한 경우 인증기관이 특정 인증 경로를 권장할 수도 있다.

이러한 권장 검증 경로는 기존의 속성인증서에 non-critical 확장영역으로 제공함으로서 기존의 체계와 호환이 가능하다. 또 다수의 권장 검증 경로를 제시하고 순차적으로 우선순위를 부여하여 여러 경로를 통한 검증을 지원할 수도 있다.(이를 위해 RVP는 SEQUENCE 구조를 가진다.)

일반적으로 체인 획득과 구성은 미리 되어 있다고 하더라도 체인의 유효성 검증은 다시 이루어져야 하므로 만약 권장 검증 경로가 문제가 있다하더라도 속성 검증자가 스스로 속성 인증서, 공개키 인증서 체인을 획득/구성/검증 하면 된다.

검증된 공개키 인증서 포리스트를 이용하여 검증할 경우 그 효율성은 인증서 체인의 중복 여부에 의존하는 데 체인의 중복이 많을수록 효율성이 증대되게 된다. 인증체계에서 상위로 올라갈수록 인증기관의 수는 적어지므로 다수의 인증서 체인은 중복된 부분을 가질 가능성이 높다. 더구나 권한 관리 기반 구조는 일반적인 공캐키 기반 구조에 비해 그 영역이 작으므로 공개키 인증서 체인의 부분적으로 중복될 확률은 더 크다. 이와 같은 검증된 공개키 인증서 포리스트를 이용한 검증 과정의 개선 이득은 포리스트를 생성/유지하는 비용을 쉽게 상쇄하고 남을 것이다.

검증된 공개키 인증서 포리스트 매우 안전한 곳에 저장되어야 하므로 저장공간에 제약을 받을 수 있다. 또 이러한 포리스트의 노드 수가 너무 많으면 유지 비용이 증가하고 노드 수가 너무 적으면 중복되는 체인을 저장하지 못하는 경우가 생기게 되어 포리스트 검색 시간만 더 걸리는 역효과를 가져오게 될 수도 있다. 따라서 포리스트의 노드 수는 이러한 효과를 생각하여 적절한 값으로 설정되어야 한다.

4. 결론

지금까지 권장 검증 경로와 검증된 공개키 인증서 포리스트를 이용하여 PMI용 인증서의 검증 시간과 노력을 줄일 수 있는 방법에 대해 설명하였다. 권장 검증 경로를 이용하여 검증에 필요한 인증서를 병렬적으로 검색/획득한 후 기존의 검증된 공개키 인증서 포리스트를 이용하여 검증 과정을 단순화하여 PMI용 인증서의 검증 시간과 노력을 줄일 수 있다.

이후에는 실제로 권장 검증 경로 확장영역의 구현과 이를 이용한 검증을 실제 환경에서 적용해 보고, 검증된 공개키 인증서 포리스트의 최적의 노드 수에 대한 연구를 진행할 예정이다.

참고문헌

[1] ITU-T, "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks" 03, 2000

[2] R. Housley, W. Polk, W. Ford, and D. Solo, "Internet X.509 Public Key Infrastructure: Certificate and CRL Profile", http://www.ietf.org/internet-drafts/draft-ietf-pkix-new-part1-12.txt,January, 2002

[3] S. Farrell, R. Housley, "An Internet Attribute Certificate Profile for Authorization", http://www.ietf.org/internet-drafts/draft-ietf-pkix-ac509prof-09.txt, 8th June 2001

[4] 이병천, 백준상, 서문석, 허원근, 김광조, PKI에서 인증트리 검증의 효율성 향상 기법, KIISC 충청지부 2000년 정보보호학술논문발표회, pp.255-261, Nov. 3~4, 2000

[5] 박 재관, 김 광조, "전체 인증경로를 알 수 있는 인증서 일련번호 부과 방법", 2001년도 한국정보보호학회 충청지부 학술발표대회, pp.292-301, 2001.10.26 -27