PKI에서 인증트리 검증의 효율성 향상 기법

*이병천, **백준상, **서문석, **허원근, *김광조

*한국정보통신대학원대학교, 정보보호그룹
**시큐아이닷컴

Efficient Verification of Certification Tree in PKI

*Byoungcheon Lee, **Joonsang Baek, **Moonseog Seo, **Weonkeun Huh, *Kwangjo Kim

*Information Security Group,
Information and Communications University
** SECUi.COM

요 약

PKI에서 인증서의 유효성을 검증하기 위해서는 인증서 자체의 검사뿐만 아니라 최상위 인증기관(Certificate Authority, CA)인 루트CA로부터 사용자에게 인증서를 발행한 단말CA까지의 인증트리의 유효성을 검증해야 한다. 그러므로 인증트리가 길어질수록 사용자가 인증서의 유효성을 검증하기 위해서 수행해야 하는 검증작업은 많은 계산량을 필요로 하게 된다. 그러나 이러한 인증트리의 확인 작업은 한 인증기관내에 속해있는 다수의 사용자들의 입장에서 보면 중복되는 작업이므로 이를 축소함으로써 효율성을 높일 필요가 있다.

본 논문에서는 이의 구체적인 구현 방법으로서, 단말CA는 자신이 속해있는 루트CA로부터 자신까지의 인증트리의 유효성을 규칙적으로 점검하고 그에 대한 증거자료를 저장하고 자신의 유효성에 관한 확인 문서(Certification Tree Validity Statement, CTVS)를 생성하여 공개하도록 한다. 이 문서는 단말CA의 유효성에 관한 공개문서(Public Witness)로서 누구나, 언제나 같은 내용을 검증할 수 있기 때문에 단말CA가 부정한 행위를 하기 어렵고 따라서 신뢰할 수 있다. 사용자는 CTVS를 참조하고 이를 신뢰하여 인증서의 유효성 검증을 빠르게 수행할 수 있으며 추후 CTVS의 문제점이 발생하면 이를 발행한 단말CA가 법적, 경제적 책임을 지게 된다.

I. 서론

공개키 기반구조(Public Key Infrastructure, PKI)[1,2,3]는 정보화 사회로 발전하기 위한 핵심 기반기술로서 전자상거래, 네트웍 보안 등 다양한 응용분야에서 공개키 암호를 안전하게 이용할 수 있게 한다. 우리나라에선 이미 전자서명법[4]이 발효되어 전자상거래의 법적인 근거가 마련되어 있으며 최근 공개키 기반구조를 구축하여 적용하려는 시도가 매우 활발하다.

현재 적용되고 있는 PKI 기술은 X.509 표준[1]에 기반하고 있고 IETF에서는 PKIX[5] 표준화 활동으로 이를 발전시켜 나가고 있다. 이것은 <그림 1>에 나타낸 바와 같이 최상위 인증기관인 루트CA로부터 단말CA 및 최종 사용자까지 계층적으로 신뢰관계가 형성되는 계층적 인증구조를 가지고 있다.

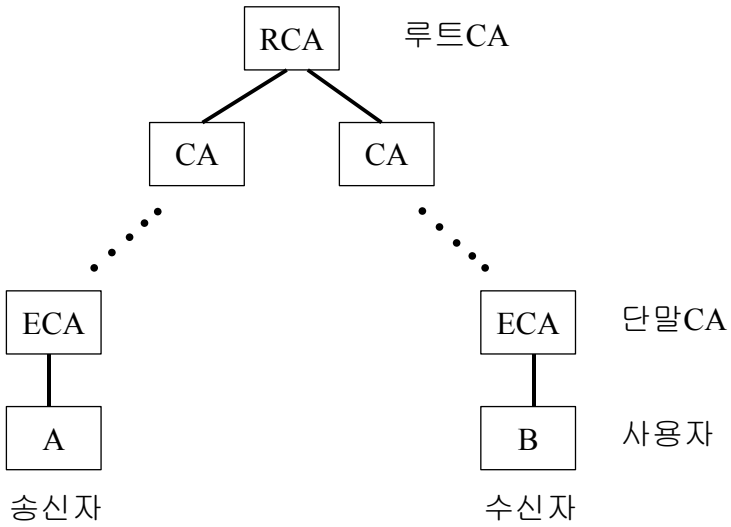

그림 1. X.509 PKI에서의 계층적 인증트리

그러므로 단말CA가 발행한 인증서의 유효성을 확인하기 위해서는 루트CA로부터 단말CA까지의 인증트리(Certification Tree)의 유효성을 검증하는 작업이 반드시 필요하다. PKIX의 RFC 2459[6] 문서에는 이러한 인증트리의 유효성 검증 알고리즘이 기술되어 있는데 루트CA로부터 단말CA까지의 인증관계를 복잡한 알고리즘을 통하여 순차적으로 모두 검증하도록 되어 있다. 그러므로 인증트리가 길어질수록 사용자가 인증서의 유효성을 검증하기 위해서 수행해야 하는 인증트리의 유효성 검증작업은 많은 계산량을 필요로 하게 되며 사용자 시스템에 큰 부담을 주게 된다. 그런데 이러한 인증트리의 유효성 검증 작업은 하나의 단말CA에 속해있는 다수의 사용자들의 입장에서 보면 중복되는 작업이므로 이를 축소함으로써 효율성을 높일 필요가 있다.

본 연구에서는 이를 해결하기 위한 방안으로서 인증서를 발급받은 다수 사용자들의 신뢰기관인 단말CA가 이러한 검증 과정을 대행하고 그에 대한 증거자료를 공개적으로 유지하며 인증트리의 유효성을 확인하는 문서(Certification Tree Validity Statement, CTVS)를 발행하도록 하였다. 사용자들을 CTVS를 신뢰하거나 직접 검증을 수행할 수 있다. 추후 단말CA가 수행한 인증트리의 검증작업이 잘못된 것으로 밝혀지는 경우에는 단말CA가 법적, 경제적 책임을 지게 된다. 이렇게 새로운 인증트리 검증방식을 이용함으로써 다수의 사용자가 똑같은 검증작업을 반복 수행하게 되는 비효율성을 줄일 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 RFC 2459에 기술되어 있는 인증트리의 검증기법을 소개한다. 3장에서는 CTVS를 생성하고 이를 이용하는 효율적인 검증기법을 기술하고 그 효과를 분석한다. 4장에서는 결론을 맺는다.

II. RFC 2459의 인증트리 검증 기법

인증트리 유효성 검증의 목적은 인증서에 기술되어 있는 사용자 정보(Subject Distinguished Name)와 사용자 공개키와의 연결관계를 모두가 신뢰하는 루트CA의 공개키에 기반하여 검증하고자 하는 것이다. 루트CA로부터 단말CA를 거쳐 최종 사용자까지는 다음과 같이 n개 실체(Entity)의 인증서 체인으로 연결되어 있다고 하자.

- o $Cert_1$: 루트CA의 자체인증서(self-signed certificate)
- o $Cert_{n-1}$: 단말CA의 인증서
- o $Cert_n$: 최종 사용자의 인증서

여기서 실체 x의 인증서는 실체 x-1에 의해 발행되며 루트CA의 자체인증서는 누구나 신뢰한다고 가정한다. 인증트리 검증 알고리즘에 입력되는 데이터는 다음과 같다.

- o 길이 n의 인증패스(Certification Path)
- o 초기정책ID(Initial Policy Identifier)의 집합
- o 현재의 날짜/시간
- o 인증패스의 유효성이 결정되는 시간 T

인증트리 검증 알고리즘은 다음과 같은 5개의 상태변수를 가지며 적절히 초기화되어야 한다.

1. Acceptable policy set : “any-policy”로 초기화
2. Constrained subtrees : “unbounded”로 초기화
3. Excluded subtrees : “empty”로 초기화
4. Explicit policy : n+1로 초기화
5. Policy mapping : n+1로 초기화

인증트리 검증 알고리즘은 $Cert_1$로부터 $Cert_n$까지 순차적으로 다음에 기술되는 검증단계들을 수행한다.

1. 기본적인 인증서 정보의 검사 : 소유자 및 발급자의 관계, 서명의 유효성, 유효기간, 인증서 취소 여부
2. Subject name이 constrained subtrees 상태변수와 일치하는가
3. Subject name이 excluded subtrees 상태변수와 일치하는가
4. 정책정보가 initial policy set과 일치하는가
5. 정책정보가 acceptable policy set과 일치하는가
6. Initial policy set과 acceptable policy set에 공통되는 set이 있는가
7. 인증서에 기재된 critical extension 수행
8. 인증서가 CA인증서인지 확인
9. permittedSubtrees에 대한 처리
10. excludedSubtrees에 대한 처리
11. Policy constraints extension에 대한 처리
12. Critical key usage extension에 대한 처리

여기서 $Cert_1$ - $Cert_{n-1}$의 CA 인증서에 대해서는 위 모든 단계를 수행하고 사용자 인증서($Cert_n$)에 대해서는 1 - 6 단계만을 수행한다. 위에 기술된 검증단계들이 인증트리 내의 모든 인증서에 대해 유효할 경우에만 사용자 인증서의 유효성이 검증되는 것이다.

III. 효율적인 인증트리 검증 기법 제안

편의상 $Cert_1$ - $Cert_{n-1}$까지의 검증을 인증트리 검증, $Cert_n$인 경우의 검증을 사용자 인증서 검증이라 하자. 즉 사용자 인증서의 유효성 검증을 위해서는 인증트리 검증과 사용자 인증서 검증의 두가지 단계를 모두 만족하여야 한다. 앞장에서 기술한 검증기법으로부터 알 수 있는 것은 사용자 인증서에 대한 검증작업을 제외하고 $Cert_1$ - $Cert_{n-1}$까지의 인증트리 검증 작업은 단말CA에 소속된 모든 사용자들에게 공통되는 작업이라는 것이다. 그러므로 사용자 시스템이 이러한 인증트리 유효성 검증 작업을 매번 직접 수행하는 것은 비효율적이며, 신뢰기관이 이를 대행하는 것이 타당하다고 생각된다. 이 장에서는 신뢰기관이 인증트리 검증작업 수행 후 CTVS를 생성하는 과정과 사용자가 이를 이용하는 방법, 그리고 문제발생시의 처리 등에 대해 기술한다. 신뢰기관은 임의의 책임있는 제3자가 될 수 있지만 여기에서는 편의상 단말CA가 그 역할을 수행하는 것으로 설정하였다.

1. CTVS 생성

단말CA는 최종 사용자 대신 인증트리의 유효성 검증을 대행함에 있어서 다음과 같은 의무사항을 가진다.

1) 단말CA는 규칙적으로 앞장에 기술된 $Cert_1$ - $Cert_{n-1}$까지의 인증트리 검증작업을 수행하고 다음과 같은 인증트리 유효성 확인 문서(CTVS)를 생성하여 게시한다.

$$CTVS_j = Sig_{ECA}(S, C, t_1, t_2, Cert_1, CRL_1, \ldots, Cert_{n-1}, CRL_{n-1}, P, CTVS_{j-1})$$

- $Sig_{ECA}()$: 단말CA(ECA)의 전자서명
- S : binary state 0 or 1 (0: valid, 1: invalid)
- C : S가 invalid인 경우의 invalid한 인증서 번호
- t_1, t_2 : CTVS의 유효기간
- $Cert_i, CRL_i$: x=i 인 인증기관의 인증서와 인증서 취소목록(CRL)
- P : 인증트리 검증 후의 policy set

2) 단말CA는 위 유효성 검증에 사용된 증거 데이터들을 누구나 접근할 수 있도록 공개 장소에 저장, 게시한다.

3) 발행된 CTVS의 잘못이 밝혀질 경우 단말CA는 법적, 경제적 책임을 지게 된다.

여기서 서명함수 내에 이전에 발행된 CTVS를 포함하는 것은 CTVS간 위조불가능한 링크를 제공하여 단말CA가 CTVS를 불법적으로 위조하는 것을 방지하기 위한 것이다.

이렇게 발행하는 CTVS는 누구나 똑같은 과정을 거쳐서 유효성을 검증할 수 있기 때문에 단말CA가 부정하게 CTVS를 발행하는 등의 행위를 하는 것은 어렵다. 그러므로 단말CA의 검증과정을 사용자가 직접 검증하지 않더라도 신뢰할 수 있으며 향후 문제가 발생한다면 사용자가 언제든지 직접 검증할 수 있고 단말CA에게 책임을 지울 수 있다.

2. CTVS의 이용

단말CA가 발행한 인증서의 유효성을 검증하기 위해서 사용자는 인증트리의 유효성과 인증서의 유효성을 함께 검증해야 한다. 송신자 A가 수신자 B에게 자신의 신분을 증명하기 위해서는 단말CA의 인증서, CRL, CTVS와 자신의 인증서를 제공하게 된다. 수신자 B는 송신자 A의 인증서의 유효성을 검증하기 위해서는 인증트리 검증과 인증서 검증을 수행해야 한다.

인증트리의 유효성을 검증하기 위해서 수신자 B는 다음 두가지 방법을 선택적으로 이용

할 수 있다.

1) 단말CA의 CTVS로부터 다음과 같이 기본적인 내용 검사 및 서명검증을 통해 인증트리의 유효성 여부를 결정한다.
   - S로부터 유효성 여부 검사
   - CTVS의 유효기간 검사
   - P의 policy set 점검
   - 단말CA의 서명 검증

2) 2장에 기술한 인증트리 검증 알고리즘을 직접 수행한다.

인증서의 유효성을 검증하기 위해서 수신자 B는 단말CA의 CRL을 검사하거나 OCSP (Online Certificate Status Protocol)[7]를 이용할 수 있다.

수신자 B는 위 두가지 검증이 모두 만족될 때에만 송신자 A의 인증서를 유효한 것으로 판정한다.

3. 문제 발생시의 해결

사용자가 전자상거래 등 네트웍 상에서 경제적 활동을 하기 위해서는 인증서를 통해 상대방의 신분을 인증해야 한다. 상대방 인증서의 유효성 검증에 있어서 단말CA가 발행한 CTVS를 신뢰하고 경제적 행위를 수행하였는데 추후 문제 발생시에는 검증 과정을 직접 다시 수행할 수 있다. 이때 단말CA가 공개, 저장해 놓은 $Cert_i, CRL_i$ 등의 데이터를 참조할 수 있다. 단말CA가 부정하게 CTVS를 발행했다는 것이 입증되면 사용자는 단말CA에게 법적, 경제적 배상을 청구할 수 있으며 단말CA는 이에 응해야 한다.

4. CTVS와 CRL의 통합운영

단말CA는 CRL을 규칙적으로 발행하여 하부 구성원의 인증서 취소 여부를 공개하는 기능을 가지고 있다. CTVS는 상부의 인증트리의 유효성을 검증하기 때문에 기능적으로는 CRL과 반대의 역할을 하지만 비슷한 방식으로 운영되므로 CRL을 발행하는 단말CA가 이를 운영하는 것이 가장 효과적이라고 생각된다. 또한 CTVS의 기능을 CRL의 extension으로 통합 구현하여 적용하는 것도 가능할 것이다.

5. CTVS 방식의 효율성 분석

단말CA에 소속된 사용자가 m명이라 하고 루트CA로부터 단말CA까지의 인증트리가 n개의 실체로 이루어졌다고 하자. 모든 사용자에 대해 1회씩 인증서 검증을 하는 경우 기존 방

법의 경우 O(mn)의 인증서 검증을 수행해야 한다. 반면 CTVS를 이용하는 경우 단말CA에 의한 O(n)의 인증서 검증과 각 사용자에 의한 O(m)의 CTVS 검증만이 필요하다.

IV. 결론

본 연구에서는 루트CA로부터 단말CA까지의 인증트리의 유효성 검증을 효율적으로 수행할 수 있도록 검증작업을 신뢰기관에게 위탁하는 방법을 제안하였다. 누구나 검증할 수 있는 작업을 모든 사용자가 중복 수행하는 것은 효율적이지 못하며 신뢰할 수 있는 기관이 이를 대행하는 것이 타당하다고 생각된다. 이러한 기능의 위탁은 향후 문제 발생시 인증트리의 유효성 검증을 누구나 재점검할 수 있기 때문에 신뢰기관이 부정한 행위를 하는 것이 어렵고 부정한 행위가 드러났을 경우 법적, 경제적 책임을 지울 수 있다.

CTVS는 루트CA로부터 단말CA까지의 상부의 인증트리의 유효성을 나타내고 CRL은 하부 사용자의 인증서 취소 여부를 나타내기 때문에 반대의 역할을 하지만 비슷한 방식으로 운영된다. 그러므로 CRL을 운영하는 단말CA가 CTVS를 함께 운영하는 것이 효율적이며 CTVS의 기능을 CRL extension으로 구현하여 적용하는 것도 가능하리라 생각된다.

참고문헌

[1] ITU-T Recommendation X.509, The Directory: Authentication Framework, 1993.

[2] 조한진, 김봉한, 이재광, “사용자 인증을 위한 공개키 기반구조 시스템 비교 분석”, CISC'98 논문집, pp. 21-32, 1998.

[3] 이병천, 김광조, “사용자 위주의 새로운 공개키 기반구조 제안”, CISC'99 논문집, pp. 47-59, 1999.

[4] 電子署名法, 法律 第5792號, http://www.mic.go.kr

[5] Public-Key Infrastructure (X.509) (pkix), http://www.ietf.org/html.charters/pkix-charter.html

[6] Internet X.509 Public Key Certificate Infrastructure and CRL Profile (RFC 2459), http://www.ietf.org/rfc/rfc2459.txt

[7] X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP (RFC 2560), http://www.ietf.org/rfc/rfc2560.txt