효율적인 분산 OCSP 시스템 설계방안

박영호, 서철, 이경현
부경대학교 전자컴퓨터정보통신공학부

Design of an Efficient Distributed Online Certificate Status Protocol

Young-Ho Park, Chul Seo, Kyung-Hyune Rhee
Division of Electronic, Computer Telecommunications Engineering,
Pukyong National University

요 약

공개키기반구조(Public Key Infrastructure, PKI)는 인터넷상에서의 안전한 전자거래를 위한 보안 기술의 핵심적인 요소이며, OCSP(Online Certificate Status Protocol)는 PKI에서 공개키 인증서의 상태를 검증하기 위한 프로토콜이다. 최근 단일 OCSP 서버의 업무 부담을 줄이고 OCSP 서버의 개인키 노출에 대한 영향을 최소화시킬 수 있는D-OCSP(Distributed OCSP) 시스템이 제안되었다. 본 논문에서는 신원기반의 암호기법을 이용하여 효율적인 D-OCSP 시스템을 설계하기 위한 방안을 제안한다.

1. 서론

최근 인터넷을 이용한 전자거래와 통신에 대해 기밀성과 무결성, 인증 그리고 부인방지 등의 보안 요구사항을 만족할 수 있는 보안기법이 절실히 요구되고 있으며, 공개키기반구조(PKI)는 이러한 보안 요구사항을 만족하기 위한 공개키 암호 기술의 핵심적인 요소이다. 사용자의 공개키에 대한 신뢰성을 제공하기 위해 인증기관(Certification Authority, CA)은 공개키 인증서를 발급하게 되며, 이 공개키 인증서는 사용자의 신원을 증명해주는 역할을 한다. 그러나 사용자의 인증서가 만료되거나 개인키가 노출되는 경우 해당 인증서가 악용되지 않도록 취소되어야 하며, 사용자들에게 해당 인증서의 상태가 알려지도록 해야 한다.

OCSP[1]는 공개키 인증서의 상태를 실시간으로 제공해 주기 위한 프로토콜이다. 사용자가 OCSP 서버에게 원하는 인증서의 상태를 질의하면 OCSP 서버는 해당 인증서의 취소유무를 OCSP 서버의 전자서명을 포함하여 알려준다. OCSP는 단지 요청된 인증서의 상태만을 알려주므로 인증서취소목록(Certificate Revocation List, CRL)[2]보다 통신비용을 줄일 수 있다.

최근에는 단일 OCSP 서버의 업무 부담을 줄이고 서비스 거부 공격이나 단일지점 오류 문제 등을 해결하기 위해 여러 개의 OCSP 서버를 이용하는 분산 OCSP 시스템이 제안되었으며[3], 분산 OCSP 시스템에서 서버의 개인키 노출에 대한 영향을 최소화시키면서 OCSP 서버에 대한 인증서를 효율적으로 관리할 수 있는 방안에 대해서도 제안되었다[4].

본 논문에서는 신원기반의 암호기법을 이용하여 기 제안되었던 분산 OCSP 시스템보다 효율적인 분산 OCSP 시스템의 설계 방안에 대해 제안한다. 2장에서는 OCSP 프로토콜과 [4]에서 제안된 D-OCSP 기법에 대해 간략히 소개하고, 3장에서 제안 방안을 기술하도록 한다. 4장에서는 제안방안을 분석하고 5장에서 결론을 맺는다.

2. 관련연구

본 장에서는 인증서 상태검증을 위한 OCSP와 [4]에서 제안된 D-OCSP 기법에 대해 간략히 소개하도록 한다.

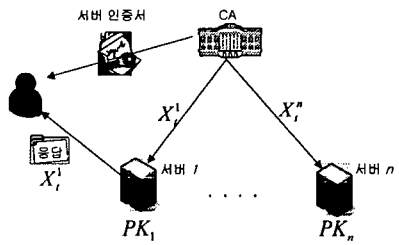
2.1 온라인 인증서 상태 검증 프로토콜(OCSP)

OCSP는 클라이언트가 인증서를 사용하고자 하는 현재시점에서 인증서의 상태를 검증하기 위한 온라인 프로토콜로서 인증서 상태에 대한 시기가 중요시되는 경우에 사용되는 프로토콜이다. 클라이언트가 OCSP

서버에게 인증서의 상태를 질의하면 서버는 인증서의 상태에 따라 "good", "revoked" 또는 "unknown"으로 응답한다. 이때 응답은 OCSP 서버에 의해 전자서명되어 전달된다. 그러나 단일 OCSP 서버를 사용하는 경우 서버의 오류에 대해 인증서의 상태를 검증할 수 없는 문제가 발생할 수 있으며, 이를 해결하기 위한 방안으로 여러 개의 OCSP 서버를 분산시키는 방법이 제안되었다.

2.2 KIS를 이용한 D-OCSP 시스템

[4]에서는 key insulated signature(KIS)[5] 기법과 [3]의 해시체인 기법을 이용하여 동일한 서명 검증용 공개키에 대해 서로 다른 서명용 개인키를 이용하여, 어느 한 서버의 개인키 노출이 다른 서버에게 영향을 주지 않도록 하면서 OCSP 서버의 인증서를 관리하는 방안을 제안하였다. 본 절에서는 [4]에서 제안된 기법의 개괄적인 동작만 살펴보도록 한다.



1. 키 생성 및 인증서 발급.

CA는 서버에 대한 마스터 비밀키 PK와 공개키 PU를 생성하고, 서버 S_i의 서명용 키를 생성하기 위해 마스터 키 PK로부터 부분키 PK_i' ($1 \le i \le n$)를 계산하여 각 서버 S_i의 서명용 개인키 PK_{S_i}를 생성한다. 각 서버는 서명을 위해 각자 소유한 PK_{S_i}를 사용하지만 클라이언트는 임의의 서버를 통해 응답을 수신하더라도 단일 공개키 PU만을 이용하여 서명을 검증한다. 키 생성과 서명에 대한 자세한 알고리즘은 [4]를 참조하기 바란다.

2. 해시체인 생성.

시스템에 n개의 OCSP 서버가 존재하는 경우, CA는 전체 주기 T에 대한 해시체인을 다음과 같이 생성.

$$X^1{}_T \rightarrow X^1{}_{T-1} \rightarrow \ldots X^1{}_t \rightarrow \ldots. \rightarrow X^1{}_1$$
$$X^2{}_T \rightarrow X^2{}_{T-1} \rightarrow \ldots X^2{}_t \rightarrow \ldots. \rightarrow X^2{}_1$$
$$\ldots.$$
$$X^n{}_T \rightarrow X^n{}_{T-1} \rightarrow \ldots X^n{}_t \rightarrow \ldots. \rightarrow X^n{}_1$$

3. CA가 OCSP 서버의 인증서 발급.

$Cert_S = Sig_{CA}(PU, SN, I, J, V, X_1^1, X_1^2, \ldots, X_1^n)$

I, J : issuer and subject of certificate

SN : serial number, V : validity period

4. 응답 메시지 검증

클라이언트는 하나의 OCSP 서버 S_i를 선택하여 인증서 상태에 대한 정보를 질의하고, S_i는 CA로부터 현재 S_i의 개인키의 유효성을 인증해주는 현재 시간주기에 대한 해시값 X_t^i ($t \in T$)를 획득하여 응답 메시지에 대한 서명과 함께 클라이언트에게 전송한다. 클라이언트는 서버의 인증서 $Cert_S$를 요청하여 인증서에 포함된 S_i에 대한 X_1^i와 응답으로 수신한 X_t^i 대해 $X_1^i = H^{t-1}(X_t^i)$ 여부를 검사하여 현재 시간주기에서 서버의 공개키에 대한 유효성을 검증하고 공개키 PU를 이용하여 서명을 검증한다.

3. 제안방안

[4]에서 제안한 기법은 클라이언트가 자신이 사용하고자 하는 인증서의 상태를 질의하고 확인하기 위해 다시 OCSP 서버의 인증서를 요구하고 이 인증서를 검증해야 과정이 필요하며, OCSP 서버의 인증서 관리를 위해 CA는 n개의 서버에 대한 해시체인을 유지해야 한다.

본 장에서는 분산 OCSP 서버의 서명과 키관리를 위해 신원기반의 암호기법을 이용하는 방안에 대해 제안한다. 신원기반 암호기법은 기존의 PKI에서 인증서 관리에 대한 부담을 줄이기 위해 Shamir에 의해 처음 제안되었으며[6], Boneh와 Franklin의 pairing을 이용한 ID-based encryption 기법[7]이 제안된 이후로 신원정보를 이용한 다양한 암호 기법들이 연구되고 있다.

제안 방안은 OCSP 서버의 응답에 대한 서명을 생성하기 위해 신원기반의 암호기법을 이용하여 OCSP 서버의 인증서에 대한 필요성을 제거할 수 있고, CA는 해시체인과 같은 상태정보를 유지할 필요가 없는 장점이 있다.

3.1 표기 및 정의

제안 방안에서는 OCSP의 서명 생성을 위해 [8]에서 제안된 신원기반의 서명기법을 적용하며, 최근 신원기반 암호기법에서 사용되는 pairing은 다음과 같은

특징을 가진다.

1) Bilinearity : $P, Q \in G_1$와 $a, b \in Z_q^*$에 대해, $e(aP, bQ) = e(P, Q)^{ab}$
2) Non-degeneracy : $e(P, Q) \neq 1$을 만족하는 $P, Q \in G_1$가 존재한다.
3) Computability : 모든 $P, Q \in G_1$에 대해 $e(P, Q)$를 효율적으로 계산할 수 있는 알고리즘이 존재한다.

신원기반 서명기법은 해당 개체의 ID가 서명검증을 위한 공개키로 이용되므로 별도의 공개키 인증서가 요구되지 않는다. 그리고 OCSP 서버는 CA의 도메인에 소속된 개체이므로 CA와 각각의 서버는 서로 신뢰한다고 가정하며, 신원정보에 대한 서명용 키는 CA가 발급한다. CA가 마스터 비밀키를 이용하여 각 서버의 ID를 이용하여 서로 다른 키를 생성하므로 KIS의 키 생성과 동일한 개념으로 생각할 수 있다[5]. 표1은 본 논문에서 사용되는 표기에 대해 설명하였다.

표 2. 표기

q	임의의 큰 소수
G_1	차수 q인 타원곡선상의 점들의 집합
G_2	차수 q인 유한체상의 곱셈군
e	paring $e: G_1 \times G_1 \rightarrow G_2$
P	G_1의 생성자
$P_{CA} \in G_1$	CA의 공개키;
$a \cdot P$	점 P에 대한 a 스칼라 곱
$x \in Z_q^*$	마스터 비밀키
T	키의 유효 사용기간
Q_{S_i}	서버 S_i의 ID_{S_i}에 대한 공개키
Y_{S_i}	서버 S_i의 개인키
H_1, H_2	암호학적 해시함수 $H_1: \{0,1\}^* \rightarrow G_1$ $H_2: \{0,1\}^* \times G_1 \rightarrow Z_q^*$
CA 공개 파라미터: $\langle G_1, G_2, e, P, P_{CA}, H_1, H_2 \rangle$	

3.1 키 생성

CA는 자신의 마스터 키 x와 클라이언트가 OCSP 서버를 식별하기 위한 정보를 이용하여 서버 S_i의 서명용 개인키 Y_{S_i}를 다음과 같이 생성하여 S_i에게 안전한 채널을 통해 제공한다.

$$Q_{S_i} = H_1(ID_{S_i} \| T)$$
$$Y_{S_i} = x \cdot Q_{S_i}$$

이때 서버의 ID로 CA 식별값과 서버의 URL 등을 이용하여 $ID = CA \| Server\_URL \| \ldots$ 형태로 구성할 수 있으며, T는 현재 키의 유효 기간에 대한 정보이다. 예를 들어, 만일 키의 주기가 하루 단위라면 $H_1(ID_{S_i} \| 20040521)$의 형태로 키의 생성을 위해 클라이언트가 인증서 상태정보를 요청한 현재의 날짜를 함께 인코딩함으로써 키의 사용을 하루 단위로 제한할 수 있다.

3.2 서명생성

OCSP 서버 S_i는 클라이언트의 질의에 응답하는 경우 임의의 난수 $r \in Z_q^*$를 선택하여 응답 메시지 m에 대한 서명 $\langle U, V \rangle$를 다음과 같이 생성하여 메시지 m과 함께 클라이언트에게 응답한다.

$$U = r \cdot Q_{S_i}, \quad h = H_2(m, U)$$
$$V = (r + h) \cdot Y_{S_i}$$

3.3 서명검증

서버로부터 $\{m, \langle U, V \rangle\}$를 응답으로 수신한 클라이언트는 메시지의 서명을 검증하기 위해 자신이 접속한 서버의 신원정보 ID_{S_i}와 CA의 공개키 그리고 현재 시간주기를 이용하여 다음과 같이 검증한다.

$$Q'_{S_i} = H_1(ID_{S_i} \| T), \quad h' = H_2(m, U) \quad (1)$$
$$e(P, V) \stackrel{?}{=} e(P_{CA}, U + h' \cdot Q'_{S_i}) \quad (2)$$

(2)식의 검증 결과가 참이라면 클라이언트가 수신한 메시지는 현재 시간대 T에 신원이 ID_{S_i}인 서버가 CA로부터 유효한 키를 발급 받아 올바로 서명한 결과이므로 수신 응답에 대한 유효성과 서버의 현재 상태를 동시에 검증할 수 있다.

4. 제안방안 분석

본 장에서는 3장에서 제안한 방안의 안전성과 성능을 분석하며, 성능의 효율성은 [4]의 D-OCSP 기법과 비교하여 분석한다.

4.1 안전성 분석

제안방안의 OCSP 서버의 응답에 대한 서명을 생성하기 위해 신원기반의 서명 기법을 적용하였다. 서명을 위한 각 서버의 개인키는 CA가 마스터 비밀키를 이용하여 서버의 신원정보 ID_{S_i}와 시간대 정보 T를 이용하여 생성하므로 CA의 마스터 비밀키가 노출되지 않는 한 어떤 서버가 키를 생성하거나 공격자가 서버의 신원정보로부터 서명용 개인키를 생성할 수 없으며, 따라서 서버의 서명을 생성할 수 없다. 그리고 키의 유효기간을 키 생성을 위한 스트링으로 함께 인코딩 하였으므로 서버가 유효기간이 지난 이전의 키를 사용하여 서명을 생성하는 경우, 3장의 식(1)에서 클라이언트의 서명 검증을 위한 시간 정보와 동기가 맞지 않게 되므로 서명 검증에 실패하게 되고 이전의 키로 서명된 서명문을 신뢰하지 않게 될 것이다.

4.2 효율성 분석

[4]에서 제안한 기법은 클라이언트가 자신이 원하는 인증서의 상태정보를 얻기 위해 서버의 서명 검증을 위한 서버의 인증서를 다시 요청하고 서버 인증서를 검증하는 과정을 거쳐야 한다. 제안기법은 신원기반 서명 기법을 사용하였으므로 서버의 서명을 검증하기 위해 서버의 인증서를 필요로 하지 않으며 서버의 URL과 같은 단지 자신이 접속한 서버를 식별할 수 있는 신원정보와 현재 시간대 정보를 이용하여 서버의 서명을 검증할 수 있다. 그러므로 서버의 인증서를 획득하기 위한 부가적인 통신처리가 요구되지 않는다.

그리고 제안기법은 인증서를 사용하지 않으므로 인증서에 포함된 CA의 서명 확인계산이 없으며, CA는 모든 개체들로부터 신뢰된다고 가정한다면, 단지 서버의 응답으로 받은 메시지에 대한 서명만 검증함으로써 응답 메시지에 대한 유효성과 클라이언트가 질의한 서버의 신뢰성을 함께 제공할 수 있다.

[4]의 기법은 서버 인증서의 시간대별 유효성을 위해 해시체인을 생성하였고 CA가 해시체인에 대한 상태정보를 보관해야 한다. 제안기법은 CA가 상태정보를 유지해야 할 필요는 없지만, 대신 CA가 요청된 서버의 서명용 키를 생성해야 하는 계산상의 부담이 생긴다. 그러나 키의 생성은 새로운 시간주기가 시작되는 시점에서 단지 한번만 수행하면 되므로 전체 프로토콜에는 큰 영향을 끼치지는 않는다.

5. 결론

오늘날 인터넷에서의 안전한 통신과 안전한 전자거래를 위해 인증서가 사용되고 있으며 OCSP는 인증서의 상태정보를 검증하기 위해 PKI에서 사용되는 기법이다. 본 논문에서는 분산 OCSP 환경을 효율적으로 구성할 수 있는 방안을 신원기반 암호기법을 적용하여 제안하였다. 신원기반 암호기법은 인증서의 사용을 필요로 하지 않으므로 인증서 상태검증을 위해 부가적으로 서버의 인증서를 요청하고 서버 인증서의 상태를 검증하는 과정이 필요 없으며, 서버의 응답에 포함된 단일 서명의 검증으로 응답에 대한 신뢰성과 서버에 대한 신뢰성을 함께 검증할 수 있다.

[참고문헌]

[1] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol-OCSP, IETF RFC 2560, 1999.

[2] R. Housley, W. Polk, W. Ford and D. Solo, Certificate and Certificate Revocation List(CRL) Profile, IETF RFC 3280, 2002.

[3] S. Micali, NOVOMODO; Scalable Certificate Validation And Simplified PKI Management, 1st Annual PKI Research Workshop, pp.15-25, 2002.

[4] S. Koga and K. Sakurai, A Distributed Online Certificate Status Protocol with a Single Public Key, International Workshop on Practice and Theory in Public Key Cryptography - PKC 2004, pp.389-401, 2004

[5] Y. Dodis, J. Katz, S. Xu and M. Yung, Strong Key-Insulated Signature Schemes, International Workshop on Practice and Theory in Public Key Cryptography - PKC 2003, pp.130-144, 2003.

[6] A. Shamir. Identity-based cryptosystems and signature schemes. In Advances in Cryptology - CRYPTO '84, LNCS 196, pp.47-53, Springer-Verlag, 1984.

[7]D. Boneh, M. Franklin, Identity-based encryption from the Weil pairing. In Advances in Cryptology - CRYPTO 01, LNCS 2139, pp.213-229, Springer-Verlag, 2001.

[8] J. Cha and J. Cheon, An Identity Based Signature Scheme from Gap Diffie-Hellman Groups, International Workshop on Practice and Theory in Public Key Cryptography - PKC 2003, pp.18-30, 2003.