情報保護學會論文誌
第13券 第6號, 2003. 12

검증자목록을 이용한 실시간 인증서 폐지 정보 전송의 설계*

이 용 준**†‡, 정 재 동**, 오 해 석***

Design of Online Certificate Revocation Information Transfer using Verifier Lists

Yong-jun Lee**†‡, Jai-dong Jung**, Hae Seok Oh***

요 약

공개키 인증서는 유효기간 이전에도 소유자의 신원정보 변경이나 개인키의 훼손과 같은 이유로 폐지가 가능하다. 인증서는 상대적으로 긴 시간의 유효기간을 가지기 때문에 폐지될 수 있는 가능성이 높다. 공개키 기반구조에서 기술적인 중요한 문제는 인증서 상태에 대한 처리에 있다. 본 논문은 금융 네트워크의 환경에서 적합한 실시간 인증서 상태 확인 메커니즘을 제안한다. 제안 방식의 특징은 검증자목록을 이용하여 실시간으로 인증서 폐지 정보를 전송하는데 있다. 이 방식은 성능에 대한 실험에서 대표적인 상태확인 메커니즘인 실시간 인증서 상태 프로토콜(OCSP : Online Certificate Status Protocol)과 동일한 현재성을 제공한다. 이와 동시에 감내하기 어려운 집중된 네트워크 전송에서 상태 확인의 부담을 줄인다.

ABSTRACT

A public key certificate may be revoked before its validity period due to causes like the owner identification information change or the private key damage. Since a certificate has long valid time relatively, it is possible to become revoked during lifetime of certificate. The main technical issue in the public key infrastructure is how to handle the status of the certificate. We propose a simple mechanism for online certificate status validation that is suited to the financial network. The characteristic of the proposed method is to broadcast certificate revocation information by using verifier list. The experimental results provide the same realtime as OCSP(Online Certificate Status Protocol). The proposed mechanism reduces the network load for certificate status validation in highly concentrated unbearable network.

keyword : *PKI, Certificate Status Validation, CRL, OCSP*

Ⅰ. 서 론

컴퓨터 네트워크는 온라인을 통한 비즈니스의 영역을 넓혀주는 중요한 기회를 제공하였으나 시스템의 서비스를 신뢰할 수 없거나 잠재적으로 사용할 수 없는 보안의 위험이 증가되었다. 실제 네트워크에서의 정보는 신뢰되지 않는 게이트웨이와 악성의 시스템을 경유하게 된다. 따라서 통신 엔티티의 신원확

* 본 논문은 숭실대학교 멀티미디어연구실의 산학연 연구결과로 수행되었음.
** 숭실대학교 대학원 컴퓨터학과(yjlee@koscom.co.kr, jjd@koscom.co.kr)
*** 경원대학교(oh@kyungwon.ac.kr)
† 주저자, ‡ 교신저자, 논문접수일 : 2003년 6월 5일, 심사완료일 : 2003년 11월 13일

인, 무결성, 부인방지, 기밀성을 제공하기 위하여 공개키 기반의 보안 기술에 대한 연구가 활발히 진행되고 있다.[1]

공개키 인증서는 발급되었을 때 제한된 사용을 기간을 가지고 있다. 개인키 분실 또는 키 소유자의 신원변경에 의해 유효기간 내에도 폐지될 수 있다.[2] 따라서 인증서의 존재뿐 아니라 폐지 여부를 결정할 수 있는 인증서 상태 확인 메커니즘이 요구된다.

인증서 상태 변경과 확인은 공개키 기반구조(PKI : Public Key Infrastructure)의 엔티티에게 네트워크 전송에 따르는 부담이 되고 있다.[3]

기존의 인증서 상태 확인하는 스키마는 오프라인 방식인 인증서 폐지목록(CRL : Certificate Revocation Lists)방식을 주로 사용해 왔다. 하지만 이 방식은 인증서를 검증하고자 할 때마다 인증서 폐지목록 전체를 다운받아야 하고 인증서 폐지목록의 크기가 커질수록 다운받아야 하는 목록의 크기가 증가함에 따라 다운받는 시간과 통신량의 부담으로 이어진다는 단점을 가지고 있다. 또한 기존의 인증서 상태 검증 방식들이 주로 주기적으로 발생하는 CRL에 기반을 두고 있기 때문에 인증서 현재 상태에 대한 시간차 문제가 발생한다.[4]

이러한 기존의 인증서 상태 확인 방법들의 문제점으로 인해 새로운 형태의 인증서 상태 검증 메커니즘이 제안되었으며 온라인 인증서 상태 검증프로토콜(OCSP)이 대표적이다. 그러나 OCSP는 인증서 상태 확인을 실시간으로 매번 처리해야 하기 때문에 많은 통신량을 발생시키는 또 다른 문제점을 가지고 있다. 따라서 통신량이 집중화된 금융거래 환경에서의 서버에 적용하기는 어렵다.[5]

본 논문은 금융거래 환경에서 검증자목록을 전송서버에 등록하여 서명자의 폐지시 해당하는 검증자, 즉 금융서비스 서버에 전송함으로써 실시간을 보장하고자 한다. 금융서비스 서버는 고객의 인증서 상태를 보유하여 사용하기 때문에 통신부하를 줄이고 상태 확인 시간을 줄이게 된다.

Ⅱ. 기존연구

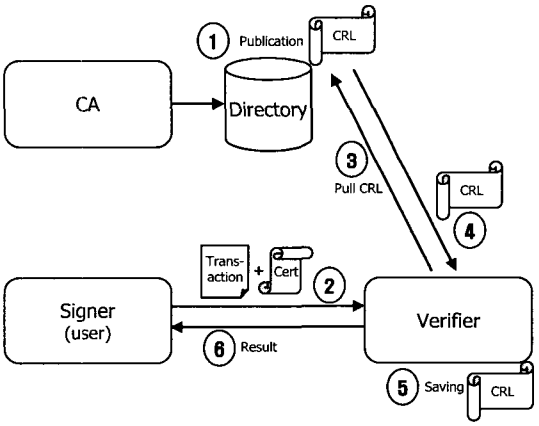
인증서는 발급 시점부터 일정기간을 사용할 수 있는데, 이를 유효기간이라 한다. 유효기간내에 개인키 분실, 자격상실, 키변경 등의 이유로 인증서를 폐지할 수 있다. 인증서 소유자는 인증기관인 CA(Certificate Authority)에 인증서 폐지를 요청하며, 인증기관은 검증자에게 인증서상태 정보를 게시한다. 이러한 과정을 통해 폐지된 인증서는 사용이 허가되지 않는다.[6]

인증서상태 검증 방법에는 대표적으로 CRL과 OCSP방식이 제시되었다. CRL은 인증기관에 의해 제시된 인증서폐지목록으로 디렉토리에 게시한다. 그러나 일정기간을 가지고 게시하기 때문에 실시간의 인증서상태 정보를 제공하지 않는다. 실시간 정보제공을 위해 OCSP 방식이 제안되었다. OCSP 클라이언트가 OCSP 서버에 인증서 검증시 매번 요청하는 방법으로 실시간 정보를 제공한다.[7]

2.1 인증서 폐지목록(CRL)

CRL은 CA가 주기적으로 폐지된 모든 인증서의 일련번호, 폐지시간, 폐지이유를 생성하여 서명한 후 디렉토리에 전송한다. 매우 간략화된 방식으로 각 사용자가 인증서 상태 확인할 때 개별적으로 접근한다. CRL은 인증서의 수가 증가함에 따라 CRL도 커지기 때문에, 많은 사용자의 통신에 적합하지 않는다. 가장 중요한 문제는 검증자가 단일 인증서를 검증할 때 폐지된 모든 인증서의 목록을 확인해야 한다는 것이다. 또한 CA에 의해 오프라인으로 처리되기 때문에 현재성에 제한을 가진다.[8]

[그림 1]은 CRL 메커니즘을 나타내었다.



〔그림 1〕 CRL 메커니즘의 인증서상태 확인 과정

<단계 ①> CA는 모든 폐지된 인증서의 목록에 전자서명을 하여 디렉토리에 게시한다.

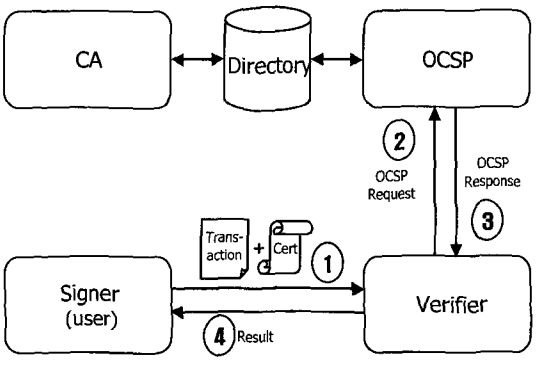
<단계 ②> 서명자는 전자서명된 거래내용을 검증자에게 전송한다.

<단계 ③> 검증자는 Pull방식으로 디렉토리에서 접속한다.
<단계 ④> 검증자는 CRL를 획득한다.
<단계 ⑤> 검증자는 재게시전까지 획득된 CRL을 저장한다.
<단계 ⑥> 검증자는 CRL을 검색한 후 거래에 대해 응답한다.

2.2 온라인 인증서 상태 프로토콜(OCSP)

OCSP서버는 클라이언트가 요청한 해당 인증서의 유효 여부를 결정하여 상태 정보를 제공한다. 이 매커니즘은 CRL보다는 실시간으로 전송하며 많은 사용자환경으로 확대가 가능하다. 그러나 클라이언트가 인증서 상태 확인할 시점에 OCSP 서버에 요청을 함으로써 네트워크에 부담이 되며 전송량이 집중화되는 문제점을 가진다. 또한 클라이언트는 OCSP 메시지를 생성하고 파싱해야 한다.[9,10]

[그림 2]는 OCSP 메커니즘을 보이고 있다.



〔그림 2〕 OCSP 메커니즘의 인증서상태 확인 과정

<단계 ①> 서명자는 전자서명된 거래내용을 검증자에게 전송한다.
<단계 ②> 검증자는 OCSP 서버에 인증서상태를 요청한다.
<단계 ③> OCSP 서버는 디렉토리를 검색하여 해당 인증서상태를 응답한다.
<단계 ④> 검증자는 OCSP 응답의 결과로써 거래에 대해 응답한다.

2.3 기존방식의 문제점

온라인 금융서비스는 거래당사자간의 권리와 의무에 대하여 상호동의 또는 상호계약이 정의되었다는 것을 의미한다. 또한 거래내용이 고부가가치를 가지기 때문에 거래당사자간의 분쟁 가능성이 존재한다. 따라서 거래내용에 대한 보안의 문제를 해결하기 위해 인증서 기반의 서비스를 제공하고 있다.

인증서기반의 금융서비스를 제공하는 대표적인 시스템은 인터넷뱅킹, 증권거래시스템, 전자상거래 등이 있다. NIST의 조사에 따르면 전체 PKI 전체 비용에서 폐지 스키마로 인하여 비용이 90%에 이르고 있다.[11] 금융서비스에서 인증서상태 확인 스키마의 구현시 고려되어야할 요구조건으로 보안, 실시간, 성능의 요구된다.[12]

- 보안(Security) : 인증기관(CA : Certificate Authority)이 보유한 인증서 상태 정보가 사용자에게 변경이 없이 획득되어야 한다.
- 실시간(Timeliness) : 인증 상태의 변경과 사용자 요청에 의한 반영 사이의 지체 시간이 최소화되어야 한다.
- 성능(Performance) : 인증서 상태 확인 스키마의 알고리즘과 프로토콜이 거래속도를 저하시켜서는 안된다.

오프라인 메커니즘인 CRL은 폐지정보에 대하여 인증기관이 전자서명을 하고 검증자는 CRL을 검증함으로써 보안이 제공된다. 또한 한번 로컬에 저장되면 CRL이 재개시되기 전까지 사용함으로써 적합한 성능을 제공한다. 그러나 CRL은 일정기간에 인증기관에 의해 생성되기 때문에 실시간 인증서상태 확인을 제공하지 못한다. 따라서 금융서비스에 적합하지 않다.[3~14]

온라인 메커니즘으로 선호되는 OCSP는 클라이언트와 서버간의 요청과 응답 메시지에 상호 전자서명을 이용한 보안과 인증서상태 확인에 대하여 실시간이 보장된다. 그러나 OCSP는 인증서상태 확인을 실시간으로 매번 처리해야 하기 때문에 많은 통신량을 발생시킴으로써 성능이 보장되지 않는다. 따라서 통

〔표 1〕 기존 메커니즘의 금융거래의 요구조건 평가

기존 메커니즘 / 요구조건	CRL	OCSP
보안	적합	적합
실시간	부적합	적합
성능	적합	부적합

신량이 집중화된 금융거래 환경에서의 서버에 적용하기는 어렵다.[15]

[표 1]은 CRL과 OCSP 메커니즘이 금융서비스에 적용되었을 때 보안, 실시간, 성능의 3가지 요구조건에 적합한지를 비교하였다.

Ⅲ. 본 론

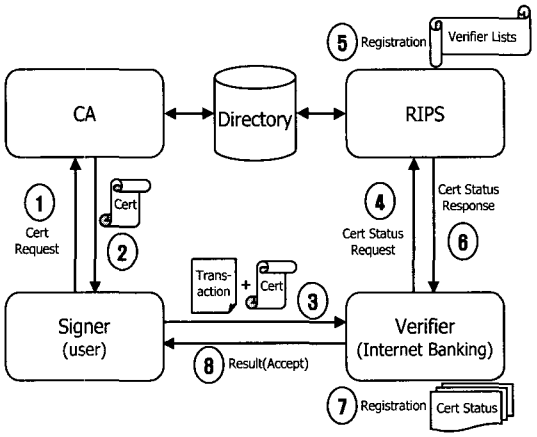
본 논문의 목적은 금융서비스에 인증서상태 확인 메커니즘을 구현할 때 요구되는 3가지의 보안, 실시간, 성능의 필수조건을 제공하고자 한다. 제안하는 알고리즘은 보안성을 제공하기 위해 응답과 요청에 대하여 전자서명과 검증을 적용하였다. 그리고 성능을 제공하기 위한 인증서상태 확인 메커니즘과 실시간을 보장하는 인증서폐지 메커니즘으로 구성된다. 제안하는 메커니즘의 구성요소는 다음과 같다.

- 인증기관(CA : Certification Authority) : 서명자에게 인증서의 발급과 디렉토리에 인증서정보의 게시를 담당한다.
- 디렉토리(Directory) : 인증기관이 발급한 인증서의 정보의 저장소의 기능을 담당한다.
- 폐지정보전송서버(RIPS : Revocation Information Push Server) : 검증자가 요청한 인증서상태에 대하여 디렉토리를 검색하여 응답을 담당하고 검증자목록에 등록하여 서명자가 폐지를 신청하면 해당 검증자에게 폐지정보를 실시간으로 전송한다.
- 서명자(Signer) : 금융서비스의 사용자로 거래내용에 대하여 전자서명을 수행한다.
- 검증자(Verifier) : 금융서비스의 제공자로 전자서명된 거래내용을 검증한다. 그 예로써, 인터넷뱅킹으로 표현한다.

3.1 인증서상태 확인 메커니즘

제안하는 인증서상태 확인 메커니즘은 OCSP보다 향상된 성능을 제공한다. 사용자가 인터넷뱅킹을 처음 사용하는 경우 인터넷뱅킹 서버는 RIPS에 인증서상태 요청하고 응답 받은 정보를 등록한다. OCSP는 모든 거래에 대해서 요청을 발생하여 성능을 저하시켰으나 제안한 메커니즘은 최초 등록시에만 요청을 하고 등록된 인증서상태 정보를 재사용함으로써 성능을 향상시킨다.

[그림 3]은 제안하는 인증서상태 확인 메커니즘을 도식화하였다.



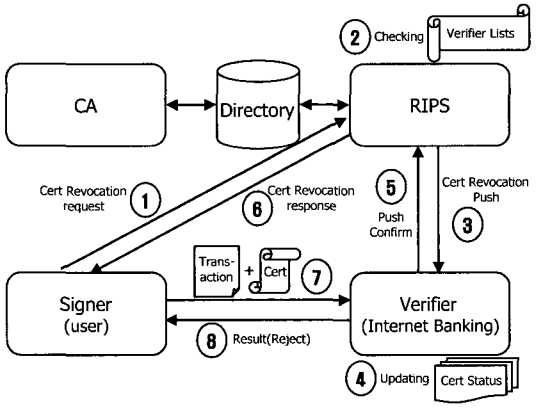
〔그림 3〕 인증서상태 확인 메커니즘

<단계 ①> 사용자는 CA에 인증서 발급을 요청한다.
<단계 ②> CA는 사용자에게 인증서를 발급한다.
<단계 ③> 사용자는 최초로써, 전자서명된 거래를 인터넷뱅킹 서버에 전송한다.
<단계 ④> 인터넷뱅킹 서버는 사용자의 인증서상태를 RIPS에 요청한다.
<단계 ⑤> RIPS는 해당 인증서의 검증자목록에 요청한 인터넷뱅킹 서버를 등록한다.
<단계 ⑥> RIPS는 인터넷뱅킹에 인증서상태 정보를 응답한다.
<단계 ⑦> 인터넷뱅킹은 해당 인증서상태 정보를 등록하여 재사용한다.
<단계 ⑧> 사용자에게 전자서명된 거래의 결과로써 승인을 응답한다.

3.2 인증서 폐지 메커니즘

제안하는 인증서 폐지 메커니즘은 OCSP와 동일한 실시간을 보장한다. 인증서상태 요청시 RIPS는 해당 인증서의 검증자목록에 요청한 인터넷뱅킹 서버를 등록한다. 사용자에 의한 폐지 요청시에 RIPS는 검증자목록을 검색하여 등록된 검증자에게 사용자 인증서가 폐지되었다는 것은 전송한다. 제안한 메커니즘은 폐지정보를 실시간으로 전송하여 OCSP와 동일한 실시간을 보장한다.

[그림 4]은 제안하는 인증서 폐지 메커니즘을 도식화하였다.



〔그림 4〕 인증서 폐지 메커니즘

<단계 ①> 사용자가 RIPS에 폐지를 요청한다.
<단계 ②> RIPS는 사용자의 검증자목록을 검색한다.
<단계 ③> 검색된 인터넷뱅킹 서버에 해당인증서의 폐지정보를 전송한다.
<단계 ④> 인터넷뱅킹 서버는 전송된 인증서 폐지정보를 변경한다.
<단계 ⑤> 인터넷뱅킹 서버는 RIPS에 인증서 폐지정보를 전송받았음을 확인한다.
<단계 ⑥> RIPS는 사용자에게 인증서 폐지를 응답한다.
<단계 ⑦> 사용자는 전자서명된 거래를 인터넷뱅킹에 전송한다.
<단계 ⑧> 인터넷빙킹 서버는 폐지정보를 전송받아 변경되었기 때문에 거래 결과로 거절을 응답한다.

3.3 인증서상태확인 메커니즘의 평가방식

인증서 상태검증 시스템은 시스템을 구성하는 각 구성요소의 부하와 각 노드간 통신량으로 그 성능을 평가할 수 있다. 기존의 CRL, OCSP 방식과 본 논문에서 제안하는 RIPS의 평가항목을 다음과 같이 나타낼 수 있다.

〔표 2〕 CRL, OCSP, 제안방안의 성능평가 항목

상태검증 방안 / 평가항목	CRL	OCSP	RIPS
CA의 연산부하	LCA	LCA	LCA
DIR/OCSP/RIPS CA간 통신량	TCA-DIR	TCA-OCSP	TCA-RIPS
DIR/OCSP/RIPS 부하	LDIR	LOCSP	LRIPS
DIR/OCSP/RIPS 검증자간 통신량	TDIR-User	TOCSP-User	TRIPS-User
검증자의 부하	LUser	LUser	LUser

[표 2]에서 L은 각 시스템의 연산부하를 의미하고, T는 전송 통신량을 의미하며 아래첨자로 표시된 기호는 각 시스템의 이름을 의미한다. 다음은 각 방식에서 평가항목을 정의하였다.

3.3.1 인증서 폐지목록(CRL) 방식의 평가항목

· CA의 연산부하 평가

CRL은 폐지된 인증서의 전자서명된 목록이다. 각 폐지인증서는 NIST 평가에 따라 68비트로 표현될 수 있다. 20비트는 일련번호를, 48비트는 폐지날짜와 시간을 표현한다. 따라서 N을 총 인증서 발행 수, P를 폐지된 인증서의 비율이라고 한다면 처리하려는 정보의 길이는 $l_{info}=68N\cdot P$ 비트이다. 폐지목록을 갱신할 때 매번 동일한 절차에 따라 전자서명되어야만 한다. 먼저 메시지 다이제스트 함수가 적용되고, 그 후에 CA의 개인키로 해쉬결과에 대하여 전자서명한다. 따라서 하루에 한번 인증서폐지목록을 갱신한다면 CA의 부하는 다음과 같다.

$$L_{CA}=T(l_{info}\cdot L_{hash}+L_{signature})$$

여기에서 다양한 암호 알고리즘이 적용되는 시스템에서 전체부하를 대표하기 위하여 두가지 기호를 도입하였다. 1-bit 메시지를 다이제스트하기 위한 연산부하 L_{hash}, 와 전자서명을 수행하기 위하여 필요한 연산부하 $L_{signature}$가 그것이다.[16]

· 디렉토리의 전송 통신량 평가

매번 갱신할 때, CRL은 디렉토리에 게시된다. 따라서 디렉토리에 게시되는 전송량은 아래와 같이 하루에 전송되는 bit 수로써 표현될 수 있다.

$$T_{CA-DIR}=T(l_{info}+l_{signature})$$

여기서 $l_{signature}$는 전자서명을 수행하는 대상 비트수이다.

· 디렉토리 연산부하 평가

CRL 기반 시스템에서는 디렉토리에 요구되는 암호연산은 없다.

$$L_{DIR}=0$$

· 디렉토리에서 검증자에게 전송되는 통신량 평가

사용자는 인증서의 상태를 확인하기 위하여 전체 CRL을 다운받아야 한다. 단지 갱신되는 주기에 따라 1회만 전송하면 재사용할 수 있으며, 다음과 같이 표현될 수 있다. Q는 하루에 요구하는 트랜잭션의 수이다.

$$T_{DIR-User} = Q(l_{info} + l_{signature})$$

· 검증자 연산부하의 평가

인증서 상태를 검증하기 위하여 사용자는 CRL을 검색하고, 그것을 검증하여야 한다.

$$L_{User} = l_{info} \cdot L_{hash} + L_{verification}$$

여기서 $L_{verification}$은 전자서명을 검증하기 위한 연산부하를 의미한다.

3.3.2 온라인 인증서 상태 프로토콜(OCSP) 방식의 평가항목

· CA의 연산부하 평가

OCSP는 CA가 인증서 상태정보를 암호연산을 거치지 않은 상태로 보관하고 있다. 따라서 CA의 연산부하는 없다.

$$L_{CA} = 0$$

· CA와 OCSP의 전송 통신량 평가

CA와 OCSP 사이의 통신량은 OCSP가 CA에 요구하는 인증서 상태정보의 양과 요구하는 회수의 곱으로 표현된다.

$$T_{CA-OCSP} = T(l_{OCSPrequest}) \cdot C_{Tr_{dir}}$$

$l_{OCSPrequest}$는 CA에 요구하는 통신 프로토콜에 따라 크기가 다르며 암호화 연산은 포함되지 않는다. Q는 하루에 요구하는 트랜잭션의 수이다.

· OCSP의 연산부하 평가

검증자가 요구하는 인증서 상태정보에 대하여 응답하기 위한 OCSP의 연산부하는 다음과 같이 표현된다.

$$L_{OCSP} = Q(l_{OCSPreply} \cdot L_{hash} + L_{signature})$$

$l_{OCSPreply}$는 OCSP 프로토콜에 따른 응답 구조체의 크기이며, $L_{signature}$는 해쉬결과에 대한 전자서명 연산부하를 나타낸다.

· OCSP에서 검증자에게 전송되는 통신량 평가

OCSP에서 검증자에게 전송되는 통신량은 OCSP 프로토콜의 응답 패킷의 길이와 전자서명의 길이를 합하여 총 트랜잭션의 수를 곱한 값이다.

$$T_{OCSP-User} = Q(l_{OCSPreply} + l_{signature})$$

· 검증자 연산부하의 평가

검증자의 연산부하는 OCSP가 전송한 응답 패킷에 대하여 해쉬연산과, 전자서명값을 검증하기 위한 연산부하를 더한 것과 같다.

$$L_{User} = l_{OCSPreply} \cdot L_{hash} + L_{verification}$$

3.3.3 제안방식(RIPS) 방식의 평가항목

· CA의 연산부하 평가

제안방식은 CA가 인증서 상태정보를 암호연산을 거치지 않은 상태로 보관하고 있다. 따라서 CA의 연산부하는 없다.

$$L_{CA} = 0$$

· CA와 RIPS의 전송 통신량 평가

CA와 RIPS 사이의 통신량은 두 가지로 표현되는데, 첫 번째는 서명자가 인증서를 폐지할 때, 서명자가 사용하였던 응용프로그램 도메인들에 PUSH하는 인증서 폐지정보와 서명자가 최초로 응용프로그램 도메인을 이용할 경우 RIPS에 인증서 상태정보를 요청하는 경우이다. 여기에서 일반 운용환경을 고려하여 최초로 서명자가 도메인을 이용하는 경우는 무시하기로 한다.

$$T_{CA-RIPS} = T(l_{RIPSrequest}) \cdot \frac{NP}{365}$$

$l_{RIPSrequest}$는 CA에 요구하는 통신 프로토콜에 따

라 크기가 다르며 암호화 연산은 포함되지 않는다. 그리고 NP/365는 하루에 폐지되는 인증서의 수이다.

· RIPS의 연산부하 평가

RIPS의 연산부하는 서명자가 인증서를 폐지하는 경우 폐지정보를 서명자가 이용하던 응용프로그램 도메인에 PUSH 하는 정보를 생산하는 부하이다.

$$L_{RIPS} = \frac{NP}{365}\ (l_{RIPSreply} \cdot L_{hash} + L_{signature})$$

$l_{RIPSreply}$는 RIPS 프로토콜에 따른 응답 구조체의 크기이며, $L_{signature}$는 해쉬결과에 대한 전자서명 연산부하이고, NP/365는 하루에 폐지되는 인증서의 수이다.

· RIPS에서 검증자에게 전송되는 통신량 평가

RIPS에서 검증자에게 전송되는 통신량은 RIPS 프로토콜의 응답 패킷의 길이와 전자서명의 길이를 합하여 총 트랜잭션의 수를 곱한 값이다.

$$T_{RIPS-User} = \frac{NP}{365}\ (l_{RIPSreply} + l_{signature})$$

· 검증자 연산부하의 평가

검증자의 연산부하는 인증서 상태정보에 대한 PUSH로 인하여 검증시점에서는 단순히 검증자 로컬에 저장되어 있는 상태정보를 읽기만 하면 되므로 연산부하는 없다.

$$L_{User} = 0$$

3.3.4 통신량 평가 시뮬레이션

실험에 앞서 위의 계산식으로 제안방식을 현재 금융거래의 환경을 가정하여 평가하여 보기로 하자. 가정하는 환경은 다음과 같다.

· 총 발행 유효인증서(N) : 4,000,000
· 폐지율(P) : 10%
· 1일 트랜잭션의 수 ($C_{Tr_{day}}$) : 100,000
· 검증자의 수 (Q) : 50
· CRL의 전체크기 (L_{info}) : 100 MB
· 전자서명한 값의 크기 ($l_{signature}$) : 128 bytes
· OCSP 패킷의 크기 ($l_{OCSPreply}$) : 1.5 KB
· 트랜잭션 하나의 크기 ($l_{OCSPrequest}$) : 0.5 KB
· 해쉬함수 연산시간 : 1 UT (Unit Time)
· 전자서명 연산시간 : 10,000 UT (Unit Time)

해쉬함수와 전자서명의 연산시간은 세 방법의 비교를 위한 것이므로 상대적인 시간으로 표현하였고, 일반적으로 1 : 10,000의 비율로 시간이 소요된다.[17]

[표 3]은 오프라인방식의 CRL에 대하여 금융환경을 고려하여 가정한 환경에서의 성능평가를 계산한 결과를 나타낸다.

[표 4]는 온라인방식의 OCSP에 대하여 금융환경을 고려하여 가정한 환경에서의 성능평가를 계산한 결과를 나타낸다.

[표 5]는 온라인방식으로 제안한 RIPS에 대하여 금융환경을 고려하여 가정한 환경에서의 성능평가를 계산한 결과를 나타낸다.

〔표 3〕 CRL 방식의 성능평가 계산결과

평가항목	계산식/계산과정	계산결과
CA 연산부하	$L_{CA} = T(l_{info} \cdot L_{hash} + L_{signature})$ = 100MB×1UT + 10,000UT	100MUT
CA-DIR 통신량	$T_{CA-DIR} = T(l_{info} + l_{signature})$ = 100MB×128B	100 MB
DIR 연산부하	$L_{DIR} = 0$	0
DIR-검증자 통신량	$T_{DIR-User} = Q(l_{info} + l_{signature})$ = 50(100MB + 128B)	5 GB
검증자 연산부하	$L_{User} = l_{info} \cdot L_{hash} + L_{verification}$ = 100MB×1UT + 10,000UT	100MUT

〔표 4〕 OCSP 방식의 성능평가 계산결과

평가항목	계산식/계산과정	계산결과
CA 연산부하	$L_{CA} = 0$	0
CA-OCSP 통신량	$T_{CA-OCSP} = T(l_{OCSPrequest}) \cdot C_{Tr_{day}}$ = 0.5KB × 100,000회	50MB
OCSP 연산부하	$L_{OCSP} = Q(l_{OCSPreply} \cdot L_{hash} + L_{signature})$ = 100,000회×(2KB×1UT+10,000UT) = 100,000 (2KUT + 10KUT)	1,200 MUT
OCSP-검증자 통신량	$T_{OCSP-User} = Q(l_{OCSPreply} + l_{signature})$ = 100,000회×(2KB + 128B)	2,128 MB
검증자 연산부하	$L_{User} = l_{OCSPreply} \cdot L_{hash} + L_{verification}$ = 2KB×1UT + 10,000UT	12 KUT

〔표 5〕 RIPS방식의 성능평가 계산결과

평가항목	계산식/계산과정	계산결과
CA 연산부하	$L_{CA}=0$	0
CA-DIR 통신량	$T_{CA-RIPS}=T(l_{RIPSrequest})\cdot\frac{NP}{365}$ $=0.5KB\times 4{,}000{,}000\times 0.1/365$	0.5 MB
DIR 연산부하	$L_{RIPS}=\frac{NP}{365}(l_{RIPSreply}\cdot L_{hash}+L_{signature})$ $=4{,}000{,}000\times 0.1/365(2KB\times 1UT+10{,}000UT)$	13 MUT
DIR-검증자 통신량	$T_{RIPS-User}=\frac{NP}{365}(l_{RIPSreply}+l_{signature})$ $=4{,}000{,}000\times 0.1/365\ (2KB+128B)$	2.3 MB
검증자 연산부하	$L_{User}=0$	0

〔표 6〕 방식의 계산성능 비교결과

구분	LCA (MUT)	TCA-DIR/OCSP /RIPS(MB)	LDIR/OCSP/ RIPS(MUT)	TDIR/OCSP/RIP S-User(MB)	LUser (MUT)
CRL	100	100	0	5,000	100
OCSP	0	50	1,200	2,128	0.012
RIPS	0	0.5	13	2.3	0

[표 6]은 CRL, OCSP, 제안한 RIPS에 대하여 성능평가를 계산한 결과를 비교하여 나타낸 것이다. 결과를 분석해 보면 RIPS는 통신량과 부하에 효율적인 것을 알 수 있으며 특히 L_{User} 항목에서 RIPS가 효율적인 성능을 보여준다.

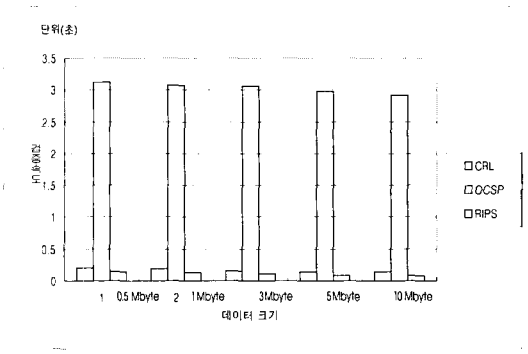
3.4 실험 및 고찰

실험환경은 시스템 하드웨어로 펜티엄 Ⅲ 800MHz, 시스템 메모리 256M SDREM으로, 운영체제는 REDHAT 7.0 리눅스이며 MY-SQL 4.0.9를 데이터베이스로 하였다. 개발언어는 C프로그램으로 gcc로 컴파일하였다.

본 실험에서는 원문의 데이터의 크기를 변화시키면서 검증자가 RIPS 서비스를 통해 해당 인증서 상태를 보유하고 검증했을 때의 속도를 분석한다. [표 7]은 실험결과를 나타낸 것이다. 제안하는 알고리즘은 금융거래에 있어 적합한 모델이기 때문에 금융거래에 있어 일반적인 통신량이 10M 내외임을 감안하여 실험을 하였다. 실험 데이터의 크기를 0.5M~10M의 범위에서 5단계로 증가시키면서 CRL, OCSP, 제안한 RIPS의 3가지에 메커니즘에 대하여 100회를 수행한 평균값이다. 실험형태는 두 가지로 분류하는데 최초등록과, 등록이후로 검증속도를 평가하도록 실험

〔표 7〕 인증서상태 검증속도 실험결과 (단위 : 초)

비교항목 \ 원문데이터	0.5M	1M	3M	5M	10M
CRL	0.14	0.15	0.16	0.19	0.21
OCSP	2.92	2.98	3.06	3.08	3.12
RIPS	0.08	0.09	0.11	0.13	0.14



〔그림 5〕 인증서상태 확인의 실험결과 분석

〔표 8〕 인증서 폐지 전송 속도 실험 결과 (단위 : 초)

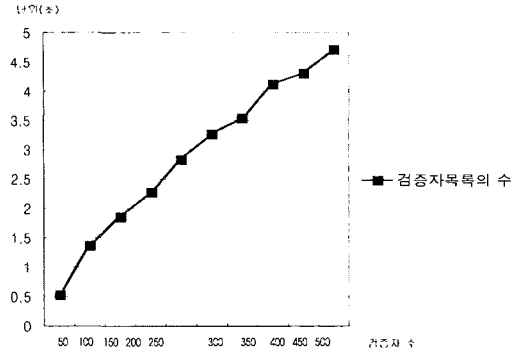
검증자수	50	100	150	200	250	300	350	400	450	500
폐지전송시간	0.53	1.37	1.86	2.28	2.84	3.27	3.55	4.12	4.31	4.72

하였다.

실험결과의 분석은 [그림 5]에서 기술한 것과 같다. RIPS 서비스를 통해 검증자가 인증서상태정보를 등록된 경우는 검증속도를 비교할 때 CRL보다 향상된 결과를 보임으로써 제안하는 방식이 성능을 개선시켰다는 결과가 도출되었다. 또한 원문의 크기가 커짐으로 CRL, OCSP, 제안한 RIPS가 다소 검증속도가 증가했으나 결과에 영향이 없음을 보여준다. 따라서 실시간 검증을 제공하는 OCSP와 비교하면 상태정보 등록 후 성능이 개선된 결과를 나타낸다.

검증자목록의 수를 50~500까지 10단계로 증가시키면서 폐지 전송속도를 측정하였다. [표 8]는 실험결과를 나타낸 것이다. 제안한 RIPS가 검증자목록의 수에 따라 폐지정보를 전송한 결과를 100회 수행한 평균값이다.

[그림 6]은 인증서 폐지 전송 속도의 실험결과를 나타낸다. 실험결과를 분석해 보면 검증자수가 적을수록 전송량이 감소하기 때문에 효율적이지만, 300개가 넘으면 OCSP보다 성능이 저하되는 결과를 나타낸다. 따라서 제안한 검증 방안은 검증자 리스트의



〔그림 6〕 인증서폐지 전송 실험 분석

수가 300개 이내일 경우 폐지전송 속도에 대한 효율성을 가진다.

Ⅳ. 결 론

본 논문은 금융서비스에 인증서상태 확인 메커니즘을 구현할 때 요구되는 3가지의 보안, 실시간, 성능의 필수조건을 제공하였다.

보안을 위해 기존의 표준을 준용하여 응답과 요청에 대하여 전자서명과 검증을 적용하였다. 향상된 성능을 제공하기 위해 제안한 인증서상태 확인 메커니즘은 최초 등록시에 상태조회를 요청을 하고 등록된 인증서상태 정보를 재사용하였다. OCSP와 동일한 실시간 보장을 위해 제안한 인증서 폐지 메커니즘은 사용자에 의한 폐지 요청시에 등록된 검증자목록을 검색하여 등록된 검증자에게 폐지정보를 전송하였다. 제안한 메커니즘은 알고리즘 평가와 실험을 통해 OCSP와 동일한 실시간을 보장하면서 향상된 성능을 제공하였다.

향후 연구과제로는 제안한 검증자목록에 대하여 폐지정보를 전송시 한계치에 대한 분석이 요구된다.

참 고 문 헌

[1] 이용준, 정재동, 오해석, "금융거래 서비스 제공자의 향상된 검증속도를 위한 인증서폐지 전송 시스템", *정보처리학회 추계학술대회*, 2002.

[2] B. Fox, B. LaMacchia. "Certificate Revocation: Mechanics and Meaning", Financial Cryptography, volume 1465, pp.158～164, February 1998.

[3] C. Adams, R. Zuccherato, "A General, Flexible Approach to Certificate Revocation", Entrust Whitepaper, June 1998.

[4] David A. Cooper, "A model of certificate revocation", Proceeding the 5th Annual Computer Security Applications Conference, December 1999.

[5] Andre' Arnes et al, "Selecting Revocation Solutions for PKI", Proceeding of NORDSED 2000 5th Nordic Workshop on Secure IT Systems, 2000.

[6] 곽진, 이승우, 조석향, 원동호 "온라인 인증서 상태 검증 프로토콜(OCSP)의 최근 연구 동향에 관한 분석", *정보보호학회지*, 2002.

[7] M. Naor, K. Nissim, "Certificate and Certificate Update", Proceeding 7th USENIX Security symposium, 1998.

[8] Housley et al, "RFC2459, Internet X.509 Public Key Infrastructure Certificate and CRL Profile", IETF, June 1999.

[9] M. Myers et al, "RFC2560, Internet X.509 Public Key Infrastructure Online Certificate Status Protocol", IETF, 1999.

[10] M. Myers et al, "Draft, X.509 Internet Public Key Infrastructure Online Certificate Status Protocol", version 2, IETF, 2002.

[11] M. Prandini, "Efficient Certificate Status Handling within PKIs: an Application to Public Administration Services", Proceedings 15th Annual Computer Security Applications Conference, 1999.

[12] John Iliadis et al, "Evaluating Certificate Status Information Mechanisms", Proceedings of the 7th ACM Conference on Computer and Communications Security, pp.1～8, 2000.

[13] R. Rivest, "Can We Eliminate Certificate Revocation List?", Financial Cryptography, Lecture Notes in Computer Science, Springer-Verlag, Vol.1465, pp.178～183, 1998.

[14] P. McDaniel, A. Rubin, "A Response to Can We Eliminate Certificate Revocation List?" Technical Report AT&T Labs, February 2000.

[15] B. Fox, B. LaMacchia. "Online Certificate Status Checking in Financial Transactions: The Case for Reissuance", Financial Cryptography, volume 1465, February 1999.

[16] W. Dai : Speed Comparison of Popular Crypto Algorithms-http://www.eskimo.com/~weidai/benchma

rks.html
[17] 최영철, 박상준, 원동호 "클라이언트-서버환경에 적합한 효율적인 인증서상태 및 경로검증 시스템", 정보보호학회 논문지, 2003.

〈著者紹介〉

이 용 준 (Yong-jun Lee) 정회원
1999년 : 강남대학교 전자계산학과 졸업
2001년 : 숭실대학교 컴퓨터학과 석사
2001년~2003 : 숭실대학교 컴퓨터학과 박사수료
<관심분야> 정보보호, 암호학, 유무선 PKI

정 재 동 (Jai-dong Jung) 정회원
1983년 : 연세대학교 수학과 졸업
1994년 : 연세대학교 산업대학원 석사
2002년 : 숭실대학교 컴퓨터학과 박사수료
1996년 : 정보통신기술사 취득(47회)
1983년~현재 : 한국증권전산 경영지원본부장
<관심분야> 정보보호, 암호학, 유무선 PKI

오 해 석 (Hai-suk Oh) 정회원
1975년 : 서울대학교 응용수학과 졸업
1981년 : 서울대학교 계산통계학과 석사
1989년 : 서울대학교 계산통계학과 박사
1976년~1982년 : 태평양화학(주), (주)삼호 전산실
1990년~1991년 : 일본 동경대학교 객원교수
1997년~1999년 : 숭실대학교 부총장
2000년~2001년 : 스탠포드대학교 객원교수
1982년~2003년 : 숭실대학교 정보과학대학 교수
2003년~현재 : 경원대학교 부총장
<관심분야> 정보보호, 멀티미디어, 데이터베이스, 영상처리