서명자의 신원정보 해쉬값을 이용한 실시간 인증서 상태 검증 메커니즘의 설계

김 현 철[†] · 김 정 재[††] · 이 종 희[†††] · 오 해 석[††††] · 전 문 석[†††††]

요 약

인증서 상태 검증 메커니즘은 공개키 기반 구조 인증 시스템의 중요한 요소이다. 현재 가장 보편적인 메커니즘은 인증서 폐지 목록을 이용하는 기법과 실시간 인증서 상태 프로토콜을 이용하는 기법이 있다. 하지만 인증서 폐지 목록을 이용하는 기법은 인증서 배포의 주기적 특성으로 인해 인증서 상태의 대한 실시간성을 보장 할 수 없으며, 실시간 인증서 상태 프로토콜을 이용하는 기법은 검증자가 검증을 요청할 때마다 중앙에 위치한 서버를 이용함으로서 서버 집중화가 발생한다. 또한 매 거래시 대량의 정보를 전송해야 하기 때문에 네트워크 과부화로 인한 인증서 상태 검증에 소요되는 시간이 오래 걸리는 문제점이 있다. 본 논문에서는 사용자의 신원정보 해쉬값을 이용하여 검증을 요청하고 인증기관과 서비스제공자가 보유하고 있는 사용자 신원정보를 통해 검증을 수행함으로써 실시간 인증서 상태 프로토콜과 같은 실시간을 보장하고 통신부하를 감소시킨다. 이에 대한 결과로 인증서 상태 검증 수행 시간을 향상 시키는 서명자 신원정보 해쉬값을 이용한 실시간 인증서 상태 검증 메커니즘을 제안한다. 또한, 실험을 통하여 기존 인증서 상태 목록 기법과 온라인 인증서 상태 프로토콜 기법에 비해 인증서 상태 검증 속도가 향상됨을 확인하였다.

키워드 : 서명자 신원정보 해쉬값, 실시간 인증서 상태 검증, 온라인 인증서 상태 프로토콜

Design of a Real-Time Certificate Status Validation Mechanism Using Identity Information Hash Value of Signer

Hyun Chul Kim[†] · Jung Jae Kim[††] · Jong Hee Lee[†††] · Hae Seok Oh[††††] · Moon Seog Jun[†††††]

ABSTRACT

The certificate status validation mechanism is a critical component of a public key infrastructure based on certificate system. The most generally mechanisms used these days are the use of the certificate revocation list and the real-time certificate status protocol. But the certificate revocation list can not give the real-time certificate status because the certificate is being delivered periodically, and the real-time certificate status protocol method will generate a concentrated load to the server because the protocol in the central server will be accessed whenever a certification is necessary. It will also take a long time to validate the certificate because each trade has to send mass information through the network. This paper will present that real-time validation is guaranteed as the real-time certificate status protocol method and the traffic congestion in the network is reduced in a way that the certification would be requested using the user information hash value and would be validated using the user information kept in the certification authorities and the service providers. Based on the this study, we suggest a real-time certificate status validation mechanism which can reduce the certificate status validation time using the signed user information hash value. And we confirm speed of certificate status verification faster than existing CRL(Certificate Revocation List) and OCSP(Online Certificate Status Protocol) method by test.

Key Words : Identity Information Hash Value of Signer, Real-time Certificate Status Verification, Online Certificate Status Protocol(OCSP)

1. 서 론

정보화 사회의 급속한 발전으로 인해 과거 오프라인으로 처리되던 많은 일련의 작업들이 온라인 처리로 급속하게 전환되고 있다. 하지만 온라인에서의 정보처리는 전송되는 정보의 노출 및 위·변조와 같은 외부 위협으로부터 항상 노출되어 있다. 이러한 위협으로부터 중요 정보를 보호하기 위해서는 기밀성(Confidentiality), 무결성(Intergrity), 부인방지(Non-Repudiation), 인증(Authentication)등의 기능을 통해 해당 거래에 대한 신뢰성과 안정성을 보장할 수 있는

† 준 회 원 : 숭실대학교 컴퓨터학과 박사과정
†† 준 회 원 : 숭실대학교 컴퓨터학과 박사
††† 종신회원 : ㈜리테일테크 연구소장
†††† 종신회원 : 경원대학교 소프트웨어대학 교수/부총장
††††† 종신회원 : 숭실대학교 정교수
논문접수 : 2005년 7월 19일, 심사완료 : 2005년 10월 17일

인증 메커니즘이 요구된다[1-3].

공개키 기반구조(Public Key Infrastructure : PKI)를 이용한 인증 시스템은 공개키 암호화 개념을 이용하여 송수신 데이터를 암호화하고 인증서를 통해 사용자를 인증하는 기술로서 디지털서명과 메시지 인증코드를 사용하여 기밀성과 무결성을 보장하고 별도의 부인방지 프로토콜을 이용하여 부인방지 기능을 제공한다. 또한 사용자인증과 메시지 인증을 통해 인증기능을 제공한다[4].

이러한 공개키 기반 구조 인증시스템은 거래가 발생 할 때 마다 해당 거래가 유효한 거래인지 아닌지를 판별하는 유효성 검증 과정을 수행한다. 유효성 검증은 전송되는 메시지에 대한 전자서명 검증과 해당 거래의 사용되는 인증서의 대한 인증서 상태 검증으로 구분 된다. 특히 인증서 상태 검증 기술은 인증서의 현재 상태와 인증서의 소유자 및 발행자의 신원을 확인하는 과정으로 전자거래에 있어 가장 중요한 부분이다. 이러한 인증서 상태 검증 기법은 인증서 폐지 목록(Certificate Revocation List : CRL)[6-7]의 주기적 특성을 이용하는 주기적 기법과 주기적인 기법의 문제점을 해결한 실시간 기법으로 구분할 수 있다[2], [5].

우선 인증서 폐지 목록을 이용하는 인증서 상태 검증 기법은 일정시간 간격으로 인증서 폐지 목록을 다운 받아 인증서 상태 검증을 수행한다. 이러한 이유로 인증서 상태에 대한 실시간성을 반영할 수 없으며, 또한 발급된 인증서의 수가 증가 할 수록 폐지되는 인증서 수 또한 비례적으로 증가하기 때문에 CRL을 저장하기 위한 대량의 저장 장치가 필요하다는 문제점이 있다[8-9].

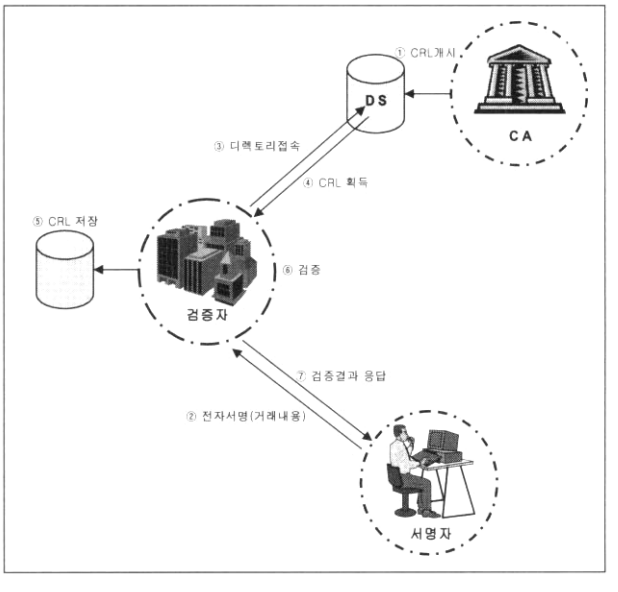
이러한 주기적 기법에 문제점을 해결하기 위한 방안으로 실시간 인증서 상태 프로토콜(Online Certificate Status Protocol : OCSP)[10-11]과 같은 실시간 상태 확인 기법이 제기되었다. 이 기법은 실시간으로 인증서 상태를 확인할 수 있다. 하지만 검증요청자가 인증서 상태 검증을 요청 할 때 마다 중앙에 위치한 서버를 이용함으로서 서버 집중화가 발생하며, 지역적인 분산 및 부과적인 시스템 도입이 필요하다는 문제가 있다. 또한 매 거래시 대량의 정보를 전송해야 하기 때문에 네트워크 과부하로 인한 처리비용 증가 및 인증서 상태 검증에 소요되는 시간이 오래 걸린다는 문제점을 가지고 있다[12].

본 논문은 실시간 인증서 상태 검증 기법의 구조적인 집중화 현상과 상태 검증을 요청할 때마다 불필요한 정보까지 모두 전송함으로서 야기될 수 있는 검증 시간에 대한 효율성 문제를 해결하기 위하여 사용자 신원정보의 해쉬값을 전송하여 인증서 상태 검증을 요청한다. 또한 CA와 서비스제공자가 보유하고 있는 사용자 신원정보를 통해 검증을 수행함으로써 인증서 상태 검증에 실시간을 보장하고 통신부하를 감소시킨다. 이에 대한 결과로 인증서 검증 수행속도를 향상시킴으로서 실시간 금융거래에 적합한 인증서 상태 검증 기법을 제안하고자 한다.

2. 관련 연구

2.1 인증서 폐지 목록(CRL)

인증서 폐지 목록은 인증기관(Certification Authority : CA)이 주기적으로 폐지된 모든 인증서의 일련번호, 폐지시간, 폐지이유를 생성하여 서명한 후 디렉토리에 게시하고 검증자는 게시된 인증서 폐지 목록을 다운 받아 인증서 상태를 검증하는 방식이다. 인증서 폐지 목록에는 게시시간, 다음 게시시간을 모두 포함하므로 이용자는 현재 갖고 있는 인증서 폐지 목록이 가장 최근 것인가를 알 수 있다. 인증기관들은 발급된 인증서들에 대한 인증서 폐지 목록을 주기적으로 갱신하여야 하며, 인증서 폐지 목록 분배를 위한 통신비용이 매우 높기 때문에 PULL 분배를 사용한다. 검증자는 인증서 폐지 목록 조회에 따른 통신비용을 줄이기 위하여 다음 변경시간까지 그 인증서 폐지목록을 보유하여 참조한다[6], [7].



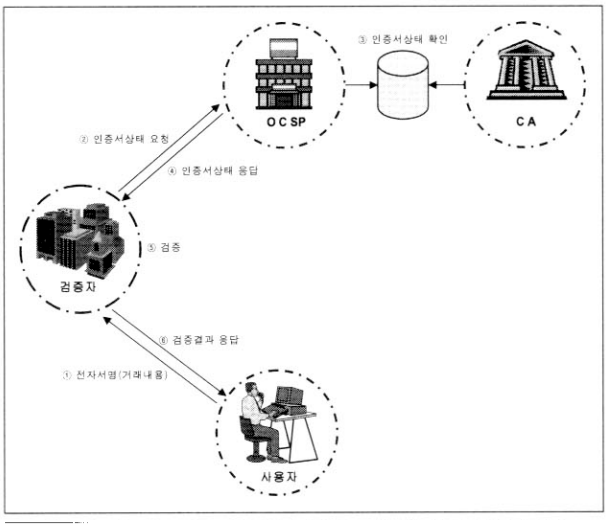
(그림 1) CRL을 이용한 인증서 상태 확인 과정

위에 (그림 1)은 인증서 폐지목록을 이용한 인증서 상태 확인 과정을 보여주고 있으며, (그림 1)의 대한 절차는 아래와 같다.

① CA는 모든 폐지된 인증서의 목록에 전자서명을 하여 디렉토리(Directory Server : DS)에 게시한다.
② 서명자는 전자서명된 거래내용을 검증자에게 전송한다.
③ 검증자는 디렉토리에 접속한다.
④ 검증자는 Pull 방식으로 CRL을 획득한다.
⑤ 검증자는 재게시전까지 획득된 CRL을 저장한다.
⑥ 검증자는 해당 인증서가 CRL에 있는지 확인한 후 전자서명에 대해 검증을 한다.
⑦ 검증자는 서명자에게 검증결과를 응답한다.

2.2 온라인 인증서상태 확인 프로토콜(OCSP)

OCSP는 주로 거래내용이 중요하여 실시간 인증서상태 확인이 요구되는 경우에 사용되며 CRL기법의 비 실시간성 문제를 해결한다. 하지만 인증서 상태 검증 요청 정보와 검증 결과 정보의 데이터가 크기 때문에 통신부하가 발생한다[10], [11], [13].



(그림 2) OCSP를 이용한 실시간 인증서상태 확인 과정

위에 (그림 2)는 OCSP를 이용한 실시간 인증서 상태 확인 과정을 보여주고 있으며, (그림 2)에 대한 절차는 아래와 같다.

① 서명자는 전자서명된 거래내용을 검증자에게 전송한다.
② 검증자는 OCSP 서버에 인증서상태를 요청한다.
③ OCSP 서버는 CA의 데이터베이스를 검색한다.
④ OCSP 서버는 검증자에게 해당 인증서상태를 응답한다.
⑤ 검증자는 인증서상태 응답을 확인한 후 전자서명에 대해 검증을 한다.
⑥ 검증자는 서명자에게 검증결과를 응답한다.

2.3 기존 인증서 상태 검증 방식의 문제점

주기적인 방식의 CRL기법은 폐지정보에 대하여 인증기관이 전자서명을 하고 검증자는 CRL을 다운받아 검증함으로써 보안이 제공된다. 또한 검증자가 CRL을 획득한 후 자신에 컴퓨터에 저장하면 CRL이 재갱신되기 전까지 사용함으로써 적합한 성능을 제공한다. 그러나 CRL기법은 인증기관에 의해 일정시간 간격으로 생성되어 배포되기 때문에 실시간으로 인증서 상태 검증을 제공하지 못한다. 더욱이 실시간 처리가 중요시 되는 분야에서는 실시간으로 인증서 상태의 대한 검증이 제공되지 않으면 거래당사자간의 분쟁 가능성이 존재하게 된다. 이러한 이유로 CRL기법은 실시간 응용 분야에 적합하지 않다는 문제점이 있다. 또한 인증서 발행 수량의 증가에 따라 폐지되는 인증서의 수량도 증가하기 때문에 폐지된 인증서의 정보를 담고 있는 CRL의 크기 또한 지속적으로 증가하게 되는 단점이 있다[8].

실시간으로 인증서 상태 검증을 처리하는 OCSP기법은 이용자와 서버간의 요청과 응답 메시지에 상호 전자서명을 이용한 보안과 인증서 상태 확인에 대하여 실시간이 보장된다. 그러나 검증자와 CA간의 검증 요청과 응답 정보 즉 OCSP Request Message와 OCSP Response Message가 담고 있는 정보는 검증자와 CA간의 통신하는데 있어서 불필요한 정보 또한 포함하고 있다. 이와 같은 이유로 통신부하가 발생할 수 있으며 검증 수행 속도 또한 효율성이 보장되지 않는다. 따라서 통신량이 집중된 클라이언트-서버 환경에서 이용하기에는 부담이 크다.

3. 제안하는 시스템

3.1 제안하는 신원정보 기반의 메커니즘

국내 공인 인증체계에서는 대면확인을 거쳐 높은 수준의 보안을 유지하고 있다. 이를 위해 사용자는 인증서 등록과정에서 개인은 주민번호, 법인은 법인번호를 CA에게 전송해야 한다. 따라서 CA는 사용자에 대한 신원정보와 인증서 상태 정보를 보유하고 있다. 또한 인터넷뱅킹, 증권거래시스템, 전자상거래 등의 온라인서비스를 사용하기 위해서 사용자는 가입의 절차를 통해 아이디와 패스워드를 부여받아야 한다. 이러한 가입과정에서 사용자의 신원정보가 서비스 제공자에게 제공하도록 되어 있다.

온라인서비스 사용자는 특정거래에 대해 전자서명을 수행하여 서비스제공자에게 전송한다. 서비스제공자는 다수의 사용자가 전송한 전자서명을 검증해야 한다. 이때 인증서상태 확인을 수행하는 과정에서 CA와 서비스제공자가 보유하고 있는 신원정보의 대한 해쉬값을 이용하여 인증서 상태 검증을 요청하고 CA와 서비스제공자는 전송받은 해쉬값에 대한 사용자 신원정보를 찾고 인증서 상태 검증을 수행하는 새로운 인증서 상태 검증 메커니즘을 제안한다.

3.2 인증서 등록 및 발급

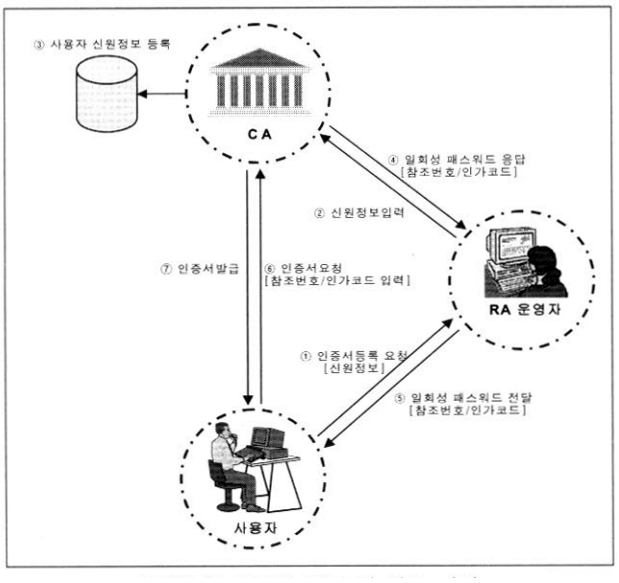
공인인증체계에서의 사용자 신원정보는 인증서를 식별할 수 있는 정보로 사용되며, (그림 3)은 인증서등록 및 발급 과정을 나타낸 것이다.

(그림 3)의 대한 절차는 아래와 같다.

① 사용자는 등록대행기관(Registeration Authority : RA) 운영자에게 인증서신청서와 신분증사본을 제출하여 인증서등록을 요청한다.
② RA운영자는 전달받은 사용자정보를 CA에 입력한다.
③ CA는 사용자의 신원정보를 데이터베이스에 저장한다. 추후 CA는 검증자의 인증서상태 요청에 대하여 저장된 신원정보에 해당하는 인증서상태를 응답한다.
④ CA는 일회용 패스워드인 참조번호와 인가코드를 RA 운영자에게 부여한다.
⑤ RA운영자는 참조번호와 인가코드를 사용자에게 전달한다.
⑥ 사용자는 인증서 관리 프로토콜(Certificate Management

Protocol : CMP)을 통해 부여된 참조번호와 인가코드 확인하고 CA에 인증서를 요청한다.

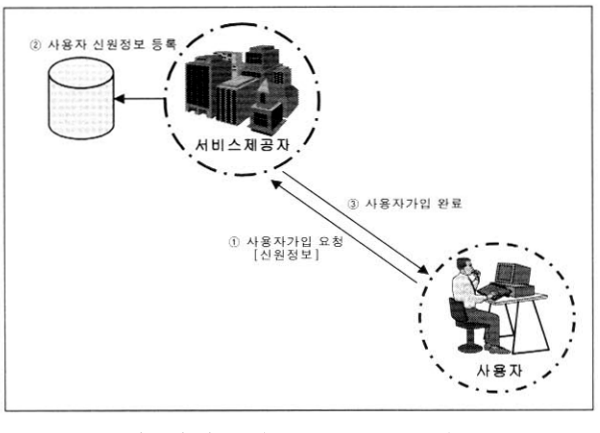
⑦ CA는 CMP를 통해 전송된 참조번호와 인가코드를 확인한 후 인증서를 발급한다.



(그림 3) 인증서 등록 및 발급 과정

3.3 온라인서비스 가입

(그림 4)는 온라인서비스 가입 과정을 나타낸 것이다. 인터넷뱅킹, 증권거래시스템, 전자상거래 등의 서비스제공자는 사용자에게 아이디와 패스워드를 부여하기 위해 사용자의 신원정보를 사용한다.



(그림 4) 온라인서비스 가입 과정

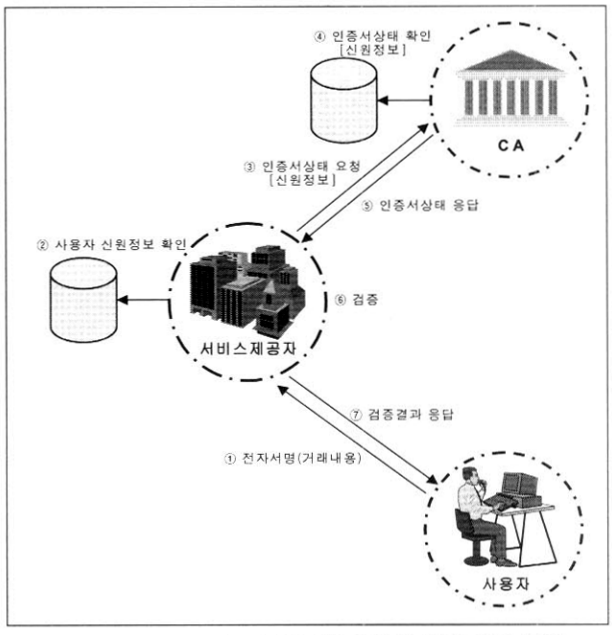
① 사용자는 서비스제공자에게 신원정보를 전송하여 가입을 요청한다.

② 서비스제공자는 사용자의 신원정보를 데이터베이스에 저장한다. 추후 서비스제공자는 사용자의 인증서상태 확인을 위해 신원정보를 사용한다.

③ 서비스제공자 접속할 수 있는 아이디와 패스워드를 사용자에게 부여한다.

3.4 신원정보 해쉬값을 이용한 인증서 상태 검증

(그림 5)는 본 논문에서 제안하는 신원정보 해쉬값을 이용한 인증서 상태 검증 과정을 나타낸 것이다. CA는 인증서등록 및 발급 과정에서 신원정보를 보유하고 있다. 또한 서비스제공자 역시 사용자 가입 과정에서 신원정보를 확보하고 있다. 제안하는 인증서 상태 검증 방식은 서비스제공자와 CA가 보유하고 있는 신원정보를 이용하여 인증서 상태 확인을 제공한다.



(그림 5) 신원정보를 이용한 인증서 상태 검증 과정

① 사용자는 온라인서비스에 접속하여 특정 전자거래에 본인의 개인키로 전자서명을 수행한 후 서비스제공자에게 전송한다.

② 전송된 전자서명의 사용자에 대하여 서비스제공자가 보유한 데이터베이스의 신원을 확인한다.

③ 신원확인을 통해 적법한 사용자 여부를 확인한 후 CA에 신원확인을 전송하여 인증서 상태를 요청한다.

④ 서비스제공자가 요청한 신원정보에 대하여 CA가 보유한 데이터베이스에 존재하는지 확인한다.

⑤ CA는 해당 사용자의 인증서 상태를 서비스제공자에게 응답한다.

⑥ 서비스제공자는 응답 받은 인증서 상태가 유효인지를 확인한 후 전자서명 검증을 수행한다.

⑦ 서비스제공자는 사용자에게 전자서명 검증결과를 응답한다.

3.5 프로토콜 정의

아래 <표 1>은 본 논문에서 제안하는 메커니즘에서 인용하는 약어의 대한 정의를 보여주고 있으며, <표 2>는 본 논문에서 제안하는 메커니즘의 프로토콜 수행 절차를 보여준다.

〈표 1〉 프로토콜 정의

Signer : 온라인서비스 사용자
Verifier : 온라인서비스 제공자
CA : 인증기관
L(Label) : 라벨, 검증자와 CA간의 통신 식별
k : 대칭키
M : 원문, 거래내용
S : 원문 M에 대한 전자서명
EK(Enveloped Key) : 대칭키 k에 대한 전자봉투
SSN(Social Security Number) : 신원정보, SSN1은 서비스제공자가 보유한 신원정보, SSN2는 CA가 보유한 신원정보
HSSN(Hashed Social Security Number) : 해쉬 신원정보
EHSSN(Encrypted Hashed Social Security Number) : 암호화된 해쉬 신원정보
CS(Certificate Status) : 인증서상태
ECS(Encrypted Certificate Status) : 암호화된 인증서상태
R(Result) : 통신결과
h( )(hash) : 일방향성의 해쉬함수
Ek( )(Encrypt) : 대칭키 k를 이용한 암호화 함수
Dk( )(Decrypt) : 대칭키 k를 이용한 복호화 함수
EX( )(Envelop) : X의 공개키를 이용한 비 대칭키 암호화 함수
DX( )(Develop) : X의 개인키를 이용한 비 대칭키 복호화 함수
SX( )(Sign) : X의 개인키를 이용한 전자서명 함수
VX( )(Verify) : X의 공개키를 이용한 검증 함수

〈표 2〉 제안하는 메커니즘의 프로토콜 수행 절차

Step	Signer (User)		Verifier (Service Provider)		CA
1			Generation L		
2			Generation k		
3			$EK = E_C(k, L)$	EK →	
4					$k, L = D_{CA}(EK)$
5	$S = S_A(M)$	S →			
6			Selection SSN_1 from DB		
7			$HSSN1 = h(SSN_1, L)$		
8			$EHSSN1 = E_k(HSSN_1)$	$EHSSN_1$ →	
9					$HSSN_1 = D_2(EHSSN_1)$
10					Selection SSN_2 from DB
11					$HSSN_2 = h(HSN_2 L)$
12					comparison $HSSN_1$ and $HSSN_1$
13					Selection CS from $HSSN_2$
14				ECS ←	$ECS = E_k(CS, L)$
15			$CS, L = D_k(ECS)$		
16			Checking CS L		
17			$V_A(S)$		
18	Comfirmation R	R ←			

① 서비스제공자는 CA와 통신을 위해 라벨 L을 생성한다. 생성된 L은 해당 서비스제공자와 CA간의 통신을 식별하는데 사용된다.

② 서비스제공자는 CA와 인증서상태 요청과 응답시의 메시지에 대한 대칭키로 사용할 k를 생성한다.

③ 서비스제공자는 라벨 L과 대칭키 k를 CA의 공개키로 암호화하여 전자봉투 EK를 CA에 전달한다. 해당 서비스제공자와 CA간의 L과 k가 안전하게 교환하여 제3자의 열람을 봉쇄한다.

④ CA는 전송된 전자봉투 EK를 CA의 개인키로 복호화하여 L과 k를 확보한다.

⑤ 사용자는 거래내용 M에 대하여 개인키로 전자서명을 수행하여 S를 서비스제공자에게 전송한다.

⑥ 서비스제공자는 전송된 S에 해당하는 사용자 신원정보 SSN1을 데이터베이스에서 획득한다.

⑦ 서비스제공자는 신원정보 SSN1과 통신식별 L을 해쉬하여 HSSN1을 생성한다. 해쉬함수는 일방향의 특성이 있기 때문에 신원정보가 외부에 공개되지 않는다. 또한 검증자와 CA간의 통신 데이터를 해쉬의 길이로 감소시킨다.

⑧ 서비스제공자는 HSSN1에 대하여 대칭키 k로 암호화를 수행하여 EHSSN1을 CA에 전송한다.

⑨ CA는 전송된 EHSSN1을 교환된 k로 복호화를 수행하여 HSSN1를 획득한다.

⑩ CA는 인증서등록과정에서 보유하고 있는 사용자의 SSN2를 데이터베이스에서 획득한다.

⑪ CA는 보유하고 있는 신원정보 SSN2와 통신식별 L을 해쉬하여 HSSN2를 생성하여 무결성 비교를 위해 사용된다.

⑫ CA는 HSSN1과 HSSN2를 비교하여 사용자의 신원정보에 대한 무결성을 검증한다.

⑬ CA는 무결성을 확인한 후 HSSN2에 해당하는 인증서 상태 CS를 데이터베이스에서 획득한다.

⑭ CA는 인증서상태 CS와 통신식별 L을 대칭키 k로 암호화하여 ECS를 서비스제공자에게 전송한다.

⑮ 서비스제공자는 전송된 ECS를 대칭키 k로 복호화하여 CS와 L을 획득한다.

⑯ 서비스제공자는 CA와의 통신식별 L과 인증서상태 CS를 확인한다.

⑰ 서비스제공자는 인증서상태 CS가 유효하면 거래내용에 대한 전자서명을 검증한다. 검증결과에 대하여 정상과 에러로 응답 R을 사용자에게 전송한다.

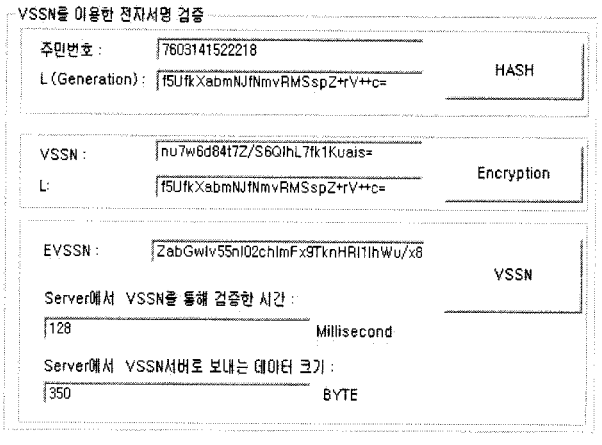
⑱ 사용자는 서비스제공자가 응답한 R에 대해 확인한다.

4. 실험 및 비교분석

4.1 실험환경

제안하는 인증서 상태 검증 시스템의 서버는 Compaq EVO W8000, CPU : Intel Xeon 2GHz, RAM : 2048MB, HDD : 36GB SCSI를 사용했으며 클라이언트는 CPU : Pentium Ⅳ 2.4MHz(슬롯), RAM 1024M HDD : 60GB를 사용했다. 시스템 소프트웨어는 서버의 운영체제로 리눅스를 사용하였으며 클라이언트의 운영체제는 윈도우XP를 사용했다. 또한 데이터베이스는 MY-SQL을 사용했으며 클라이언트 개발환경은 Visual C++ 6.0과 ActiveX, SSL을 이용하였다. (그

림 6)은 본 논문에서 제안하는 인증서 상태 검증 메커니즘의 실험 화면을 보여주고 있다.



(그림 6) 제안하는 인증서 상태 검증 실험 화면

<표 3>은 기존의 인증서상태 확인 프로토콜인 CRL기법과 OCSP기법 그리고 본 논문에서 제안한 기법을 비교 실험한 데이터를 명시하였다. 각각의 실험에서 사용한 인증서는 세 가지 기법 모두 국내 공인 인증기관에서 발급한 인증서를 사용하였다. CRL기법의 실험은 검증자가 최초에 갱신된 CRL을 디렉토리에서 획득하는 과정과 획득한 이후 로컬에서 검증하는 과정으로 나누어서 실험하였다.

실험결과에서 CRL기법의 경우 디렉토리에서 획득하고 확인할 때 부담이 있지만 획득한 이후에는 효율적인 것으로 나타났다. 그러나 실시간 보장이 되지 않는다.

OCSP기법은 요청자가 OCSP Request에 전자서명을 하였기 때문에 데이터크기가 증가하고 수행시간 역시 부담이 있는 것을 보여주고 있다. 제안하는 메커니즘은 OCSP와 동일하게 실시간을 보장하고 전송량과 수행시간에서 있어서 감소되는 결과를 확인할 수 있다.

〈표 3〉 인증서 상태 검증 실험 결과

분류	전송량	수행시간
CRL DS 참조	3758 byte	491 ms
CRL Local 참조	0 byte	40 ms
OCSP	1948 byte	350 ms
신원확인 응용	350 byte	128 ms

4.2 비교분석

<표 4>는 본 논문에서 실험한 인증서상태 검증 기법과 기존의 인증서 상태 검증 기법간의 실험을 통한 비교 분석 결과를 보여주고 있다.

실시간성에 있어서 CRL기법은 offline이며 OCSP기법과 제안방식은 online을 보장한다. 통신량을 비교하면 CRL기법은 목록이 계속해서 증가하기 때문에 데이터크기가 high로 분류된다. OCSP기법은 CRL기법보다 데이터크기가 감소되어 있기 때문에 medium으로 분류된다. 제안방식은 신원정보에 대한 해쉬값을 사용함으로써 크기를 감소시켰으

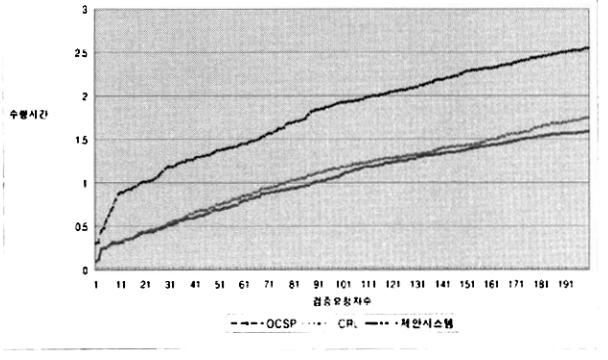
〈표 4〉 비교분석

평가항목 / 비교대상	실시간성	통신량	통신횟수	수행시간	비고
CRL	offline	high	low	고속	전자결제, 의료정보시스템
OCSP	online	medium	high	저속	인터넷뱅킹, 전자상거래
신원정보 응용	online	low	high	고속	증권거래, 전자입찰

며 low로 분류된다. 통신횟수에 있어서 CRL은 CA가 갱신 전까지 검증자 로컬에서 사용하기 때문에 통신량은 현저히 낮으며 low로 분류된다. OCSP기법과 제안방식은 실시간으로 인증서상태를 확인하기 때문에 통신량은 high로 분류된다. 수행시간을 비교하면 CRL기법은 CRL획득 이후에는 고속으로 처리되며 OCSP기법은 요청과 응답의 데이터크기로 인하여 저속으로 처리된다. 제안방식은 요청과 응답의 크기를 최소화시킴으로써 고속의 결과를 나타내었다.

실험결과를 기초로 응용프로그램에 적합한 인증서상태 확인 방식을 분류하면 다음과 같다. CRL기법의 경우, 획득한 이후는 효율적이나 실시간이 보장되지 않기 때문에 금융거래 프로그램에서는 적합하지 않다. 그러나 전자결제, 의료정보시스템과 같이 사원 또는 의료인의 이직이 빈번하지 않고 사안이 긴급하지 않는 응용프로그램에 적합하다. OCSP기법은 실시간을 보장하나 통신량과 통신횟수를 고려할 때 확인시간이 저속인 단점이 있다. 그러나 OCSP기법은 요청과 응답의 토큰 안정성은 확보된다. 따라서 시간적 특성 보다는 안전성이 요구되는 인터넷뱅킹, 전자상거래와 같은 금융거래 프로그램에 적합하다. 제안 메커니즘은 실시간을 보장하며 통신량과 통신횟수를 고려할 때 요청과 응답의 데이터크기를 감소시켜 고속의 특성을 나타낸다. 그러나 OCSP 기법보다는 요청과 응답 과정의 안전성이 우수하지 않다. 따라서 안정성 보다는 시간적 특성이 고려되는 증권거래, 전자입찰의 금융거래 프로그램에 적합한 특성을 보인다.

(그림 7)은 기존의 CRL기법, OCSP기법 그리고 제안하는 기법에 대한 인증서 검증 수행 시간 실험결과를 보여주고 있다. (그림 7)에서 상단의 추세선은 기존의 OCSP기법의 대한 실험 결과를 보여주고 있으며 비교 실험한 다른



(그림 7) 인증서 상태 검증 실험 결과

기법에 비해 수행시간이 다소 오래 소요됨을 확인할 수 있다. 가운데 보여지는 추세선은 CRL기법의 실험결과로서 인증서 상태 목록을 검증서버에 다운 받아 인증서 상태 검증을 수행하기 때문에 OCSP기법보다 수행속도가 효율적임을 확인할 수 있다. 가장 하단에 추세선은 본 논문에서 제안하는 기법으로 인증서 상태 검증 요청시 불필요한 정보를 제거하고 사용자의 신원정보 해쉬값만을 이용하여 인증서 상태 요청과 검증을 수행하기에 다른 기법에 비해 검증 수행시간이 적게 소요됨을 확인 할 수 있다.

5. 결 론

온라인에서의 모든 정보 교환은 중요 정보의 노출 및 위·변조 등의 외부 위협요소로부터 항상 노출되어 있다. 이와 같은 위협 요소로부터 해당 거래에 대한 정당성을 확보하기 위한 수단으로 PKI 기반의 인증서를 이용한 시스템이 제안되었고 현재 거의 모든 전자상거래에서 이용되고 있다. 하지만 인증기관으로부터 발급된 인증서는 개인키 분실, 자격상실, 키 변경 등의 여러 가지 이유로 인증서 폐지가 발생하기 때문에 검증자는 수신한 인증서에 대해 상태를 확인해야 한다. 이를 위해 CRL기법과 OCSP기법이 제안되었다. 하지만 CRL기법은 실시간을 보장하지 못하고 OCSP기법은 통신량이 크기 때문에 통신부하가 발생하는 문제점이 지적되었다. 제안하는 방식은 인증서 상태 확인과정에서 CA와 서비스제공자가 보유하고 있는 신원정보를 이용한 방식을 제안한다. 제안하는 방식은 OCSP기법과 같은 실시간을 보장하고 신원정보의 해쉬값으로 인증서상태 요청을 함으로써 통신부하를 감소시킨다. 따라서 시간적특성이 고려되는 증권거래, 전자입찰의 금융거래 프로그램에 적합할 것으로 판단한다.

참 고 문 헌

[1] 권태경, 강명호, 김승주, 서정욱, 진승헌, "정보보호표준개론", 한국정보통신기술협회, pp.10~12, 2002

[2] 김현철, 안재명, 이용준, 오해석 " 축약 서명 기반의 실시간 인증서 상태 검증 기법", 정보처리학회논문지C, Vol.12-C No.02 pp.0301~0308, 2005

[3] 최연희, 박미옥, 전문석 "CA를 인증 경로 처리 작업에 참여시키는 새로운 인증서 검증 방안", 정보처리학회논문지C, Vol. 11-C, No.01 pp.0021~0030, 2004

[4] 칼리슬 아담스, 스티브 로이드 공저 | 장기식 역, "보안을 위한 효율적인 기법 PKI", pp.51~67, 2003

[5] 정재동, "CSMP 기반의 실시간 인증서 상태검증의 성능개선", 숭실대학교 박사학위 논문, pp.30~55, 2003.

[6] RFC 2459, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", 1999.

[7] RFC 3080, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List(CRL) Profile", 2002.

[8] 장홍종, 이성은, 이정현, "DARC 기반에서의 실시간 인증서 유효성 검증에 관한 연구", 정보처리학회논문지C, 8(5), pp.0517~0524, 2001.

[9] 정재동, 오해석, "실시간 인증서 상태 검증의 성능개선", 정보처리학회논문지C, Vol.10-C, No.04, pp.0433~0440, 2003.

[10] RFC 2560, "Internet X.509 Public Key Infrastructure Online Certificate Status Protocol(OCSP)", 2001.

[11] draft-ietf-pkix-ocsp2-ext-01, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol version2", 2002.

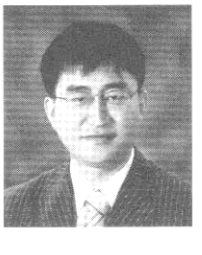
[12] 이승우, 곽진, 조석향, 주미리, 원동호, "실시간 인증서 검증 시스템 모델에 관한 연구", 한국정보처리학회 2002년 춘계학술대회논문집, 9(1) pp.0833~0836, 2002.

[13] 곽진, 이승우, 조석향, 원동호, "온라인 인증서 상태 검증 프로토콜(OCSP)의 최근 연구 동향에 관한 분석", 한국정보보호학회 학회지, pp.50~61. 2002.

김 현 철

e-mail : dmzpolice78@korea.com
2003년 인제대학교 정보컴퓨터학부(학사)
2005년 경원대학교 전자계산학과(석사)
현 재 숭실대학교 컴퓨터학과 박사과정
관심분야 : PKI, DRM

김 정 재



e-mail : argniss@empal.com
1995년 영동대학교 컴퓨터공학과(학사)
1999년 숭실대학교 컴퓨터학과(석사)
2005년 숭실대학교 컴퓨터학과(박사)
관심분야 : 멀티미디어 보안, DRM, Multimedia, Database

이 종 희

e-mail : ejonghee@retailtech.co.kr
1998년 한밭대학교 컴퓨터공학과(학사)
2000년 숭실대학교 컴퓨터학과(석사)
2004년 숭실대학교 컴퓨터학과(박사)
2004년~현재 (주)리테일테크 연구소장
관심분야 : RFID/USN, RFID 보안, EPC 보안

오 해 석

e-mail : oh@kyungwon.ac.kr
1975년 서울대학교 응용수학과(학사)
1981년 서울대학교 계산통계학과(석사)
1989년 서울대학교 계산통계학과(박사)
1982년~2003년 숭실대학교 컴퓨터학부 교수/부총장(역임)
2003년~현재 경원대학교 소프트웨어대학 교수/부총장
관심분야 : Multimedia, Database, 지식경영

전 문 석

e-mail : mjun@computing.ssu.ac.kr
1981년 숭실대학교 전자계산학과(공학사)
1986년 University of Maryland Computer Science(공학석사)
1989년 University of Maryland Computer Science(공학박사)
1989년 Morgan State University 조교수
현 재 숭실대학교 정교수
관심분야 : 전자상거래 보안, 인터넷 보안, 멀티미디어 보안