# 어플리케이션의 독립성을 위한 PKI 에이전트의 설계

이용준*, 오동열*, 정재동*, 오해석*
*숭실대학교 대학원 컴퓨터학과
mail:yjlee@koscom.co.kr

# Design of PKI agent for independence of application

Yong-Jun Lee*, Dong-yul Oh*,
Jea-Dong Jong*, Hea-Suk Oh*
*Dept of Computer Science, Han-Kook University
*Dept. of Computing, Graduate School, Soongsil University

요 약

PKI(Public Key Infrastructure)기반의 발전으로 인터넷뱅킹, 증권거래시스템, 전자메일, 전자입찰, 전자민원 등 신원확인이 요구되는 어플리케이션에 전자서명과 암호가 적용되고 있다. 각 어플리케이션은 라이브러리를 호출하여 전자서명과 검증, 암호와 복호를 수행한다. PKI기반의 어플리케이션 개발자는 상이한 인증, 암호 API(Application Programming Interface)를 호출해야 하며, 이는 프로그램의 복잡도를 증가시킨다. 개발언어와 환경에 따라서 상의한 라이브러리를 사용해야 한다. 제안하는 PKI 에이전트는 개발언어와 어플리케이션에 독립적으로 인증, 암호기능을 수행하고 결과만을 리턴한다. 따라서 어플리케이션은 인증과 암호가 필요한 시점에 검증 에이전트를 호출하게 됨으로써 프로그램의 복잡도를 줄이고 어플리케이션의 안정성을 향상시킨다.

## 1. 서론

정보보호를 위해 인증서를 기반으로 보안 메커니즘을 제공하는 기반 구조인 PKI(Public Key Infrastructure)에 대하여 많은 연구가 진행되고 있다. PKI가 정보보호 핵심 기반구조로 활용되면서 PKI 시장의 규모가 확장되고 있지만, PKI에는 앞으로 해결해야 할 문제를 여러 가지 안고 있으며, PKI 응용 어플리케이션의 인증, 암호의 통합화는 필수불가결한 상황이다.

사용자의 공개키를 안전하고 신뢰성 있게 전달하는 방법을 제공하는 PKI는 정보보호기술의 핵심이며 인증(Authentication), 무결성(Integrity), 부인방지(Non-repudiation), 기밀성(Confidentiality), 가용성(Availability)의 기능을 제공한다[1].

전자서명은 개인키의 소유자만이 생성할 수 있다. 검증은 전자서명의 진위여부를 확인하는 과정이며 개인키에 합치하는 공개키의 획득하고 전자서명값에 대한 검증을 수행한다. 공개키 획득은 인증서유효성 검증을 통해서 이루어지게 되는데 인증서에 공개키를 가지고 있기 때문이다. 사용자의 개인키 유출, 분실, 자격변경, 키변경 등의 이유로 인증서 폐지가 가능하며 검증자는 수신한 인증서의 상태가 유효한 것인지를 확인해야 한다[2]. 인증서상태 확인을 위하여 CRL(Certificate Revocation List)[3], OCSP(Online Certificate Status Protocol)[4], SCVP(Simple Certificate Validation Protocol)[5]의 제안되었다.

현재 많은 보안관련 업체에서는 기존 온라인 서비스에 PKI의 기능을 제공하여 개발하고 있다. PKI(Public Key Infrastructure)기반의 발전으로 인터넷뱅킹, 증권거래시스템, 전자메일, 전자입찰, 전자민원 등 신원확인이 요구되는 어플리케이션에 전자

서명과 암호가 적용되고 있다. 각 어플리케이션은 라이브러리를 호출하여 전자서명과 검증, 암호와 복호를 수행한다. PKI기반의 어플리케이션 개발자는 상이한 인증, 암호 API(Application Programming Interface)를 호출해야 하며, 이는 프로그램의 복잡도를 증가시킨다. 이러한 문제는 각 어플리케이션과 인증, 암호기능이 독립적이지 않기 때문이다.

개발언어와 환경에 따라서 상의한 라이브러리를 사용해야 한다. 본 논문의 검증 에이전트는 서버환경이 서버-클라이언트 또는 웹 어플리케이션에 독립적으로 인증, 암호 기능을 담당한다. 따라서 어플리케이션은 인증과 암호가 필요한 시점에 검증 에이전트를 호출하게 됨으로써 프로그램의 복잡도를 줄이고 어플리케이션의 안정성을 향상시킨다.

본 논문의 구성은 다음과 같다. 2장에서는 PKI기반 어플리케이션에 대해 분석한다. 3장에서는 어플리케이션에 독립성을 제공하는 PKI 에이전트를 제안한다. 4장에서는 기대효과를 제시한다. 5장에서는 결론을 맺는다.

## 2. 관련연구

금융거래와 증명서비스를 제공하는 인터넷뱅킹, 증권거래시스템, 전자입찰, 전자민원은 보안기능을 제공하기 위해 PKI가 적용이 의무화되었다[7]. 본 장에서는 PKI기반 어플리케이션의 특성을 분류하고 어플리케이션과 인증, 암호기능의 독립성에 대한 문제점을 제시한다[6].

### 2.1 PKI기반 어플리케이션

신원확인과 암호를 목적으로 하는 안전한 어플리케이션의 구현이 가능하다. 전자서명과 검증, 암호화 복호는 어플리케이션의 특성에 따라서 수행된다[7].

<표 1> PKI기반 어플리케이션

| 어플리케이션 | 내 용 |
|---|---|
| 인터넷뱅킹 | 이체, 예금 및 금융거래 |
| 사이버트레이딩 | 증권거래시스템 |
| 전자상거래 | 온라인 상거래 |
| 민원행정 | 각종 정부 민원서류 |
| 전자결제 | 결제시스템 |
| 보안메일 | 사용자간 보안 이메일 |
| 전자계약 | 기업간 계약문서 |

### 2.2 인증기술

전자서명은 전자문서나 메시지를 보낸 사람의 신원확인과 그 내용이 전송중에 변조되지 않았다는 무결성을 제공하기 위해 사용된다. 전자서명에서는 해당 문서의 기밀성은 제공하지 않는다. 단지 메시지의 변조 여부와 송신자의 신원만 확인하는데 목적이 있다. 전자서명을 사용함으로써 부가적으로 부인방지가 가능하게 된다. 전자서명은 공개키 암호 알고리즘에 기반을 두고 있는데, 개인키로 서명을 하고 공개키를 사용해서 서명검증을 한다[8].

### 2.3 보안기술

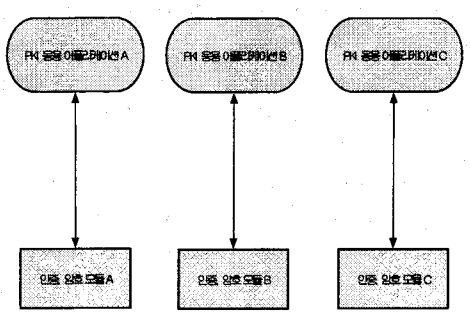
통신하고자 하는 쌍방이 암호키를 공유하고 있을 경우, 암호 알고리즘을 쌍방이 같이 선택한 후, 그 키를 사용하여 보내고자 하는 메시지를 암호화하여 보낸다. 만일 키를 공유하고 있지 않을 경우에는 별도의 키 교환 프로토콜을 사용하거나 또는 공개키 암호 알고리즘을 사용한다.

## 3. 제안하는 PKI 에이전트

본 논문에서는 대규모의 검증이 기존 방식에 대비하여 복잡도를 감소시키기 이루어지는 서버에서 안정성을 보장하고 위해 검증 에이전트를 제안한다.

### 3.1 기존 전자서명과 검증 적용방식

TCP/IP기반의 서버-클라이언트 어플리케이션의 경우는 윈도우환경에는 DLL(Dynamic Link Library)를 링크하여 전자서명 또는 검증이 필요한 경우에 호출을 하는 방식을 사용한다. 유닉스환경에서는 정적 라이브러리와 동적 라이브러리를 링크하여 사용한다.

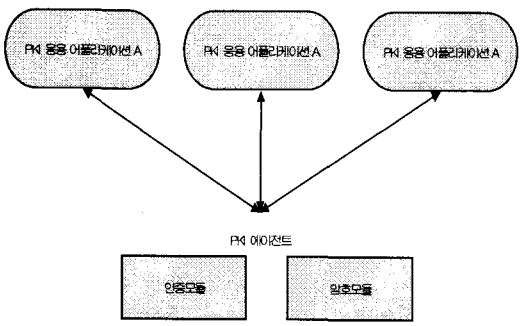



[그림 1] 기존 인증, 암호 구조

HTTP기반의 웹 어플리케이션에서는 ActiveX를 이용하여 전자서명 API를 호출하고 서버에서는 Java의 경우 Class와 Jar형식의 라이브러리를 사용하거나 ASP는 DLL의 API를 호출한다.

### 3.2 PKI 에이전트의 구성도

제안하는 PKI 에이전트는 클라이언트-서버 또는 웹 어플리케이션이 요청하는 인증과 암호 기능을 담당한다. 어플리케이션이 전자서명의 생성과 검증, 암호와 복호를 PKI 에이전트에게 요청하면 인증 기능을 수행하고 그 결과를 응답한다. [그림 2]는 PKI 에이전트의 구조를 명시한다.




[그림 2] PKI 에이전트의 구조

### 3.3 검증 에이전트의 기능

● 전자서명

공인인증서의 개인키를 이용하여 원문에 대한 전자서명을 수행한다.

● 검증

전자서명 데이터에 대하여 인증서 유효성검증과 전자서명검증을 수행한다.

● 암호화

비대칭키, 대칭키 방식으로 데이터와 파일에 대하여 암호를 수행한다.

● 복호화

비대칭키, 대칭키 방식으로 암호화된 데이터와 파일에 대하여 복호화를 수행한다.

### 3.3 시스템 통합 관리

서버-클라이언트와 웹 어플리케이션이 통합되어 관리하는 환경에서 기존의 암호와 검증 API호출 방식을 적용 하면 프로그램 복잡도가 증가한다. 이 때문에 통합관리에 부담이 되는데 PKI 에이전트는 인증, 암호 요청과 응답으로 통신하기 때문에 통합관리가 가능하다.

## 4. 기대효과

기존의 방식은 어플리케이션과 인증과 암호기능이 종속적인 것에 반하여 제안한 PKI 에이전트는 독립적인 특징을 가지고 있다. 프로그램의 복잡도 측면에서 PKI 에이전트는 인증, 암호 요구시 마다 API를 호출하는 기존 방식에 비해 요청과 응답으로 간략화되었다. 개발언어에 있어서도 PKI 에이전트와 기존 어플리케이션의 통신 프로토콜만 정의해 주면 되기 때문에 독립적이다. 서버-클라이언트와 웹의 통합관리의 제공의 면에서도 우수하다. 따라서 제안한 PKI 에이전트는 <표 2>에서 기술한 것과 같은 기대효과를 나타낸다.

<표 2> 제안하는 PKI 에이전트와 기존방식의 비교

| | 기존 검증 방식 | 검증 에이전트 |
|---|---|---|
| 프로그램 복잡도 | 높음 | 낮음 |
| 개발언어 독립성 | 종속적 | 독립적 |
| 시스템 통합 | 보장안됨 | 보장 |
| 장애처리 | 복잡 | 간략 |
| 검증속도 | 고속 | 저속 |

## 5. 결론

PKI(Public Key Infrastructure)기반의 발전으로 인터넷뱅킹, 증권거래시스템, 전자메일, 전자입찰, 전자민원 등 신원확인이 요구되는 어플리케이션에 전자서명과 암호가 적용되고 있다. 각 어플리케이션은 라이브러리를 호출하여 전자서명과 검증, 암호와 복호를 수행한다. PKI기반의 어플리케이션 개발자는 상이한 인증, 암호 API(Application Programming Interface)를 호출해야 하며, 이는 프로그램의 복잡도를 증가시킨다. 개발언어와 환경에 따라서 상의한 라이브러리를 사용해야 한다. 제안하는 PKI 에이전트는 개발언어와 어플리케이션에 독립적으로 인증, 암호기능을 수행하고 결과만을 리턴한다. 따라서 어플리케이션은 인증과 암호가 필요한 시점에 검증 에이전트를 호출하게 됨으로써 프로그램의 복잡도를 줄이고 어플리케이션의 안정성을 향상시킨다.

## 참고문헌


[1] Vishwa Prasad & Sreenivasa Potakamuri & Michael Ahern. "Scalable Policy Driven and General Purpose Public Key Infrastructure(PKI)", IEEE, 2000.

[2] Ray Hunt. "PKI and Digital Certification Infrastructure", IEEE, 2001.

[3] RFC2459, Certificate and CRL Profile, 1999.

[4] RFC2560, Online Certificate Status Protocol, 2001.

[5] Draft, Simple Certificate Validation Protocol, 2002.

[6] Andre Arnes, Svein J. Knapskog. "Selecting Revocation Solutions for PKI", NORSEC 2000.

[7] Albert Levi & M. Ufuk Caglayan. "An Efficient, Dynamic and Trust Preserving Public Key Infrastructure", IEEE, 2000.

[8] RFC2528, Representation of Key Exchange Algorithm Keys in Internet X.509 Public Key Infrastructure Certificates, 1999.