제33회 한국정보처리학회 춘계학술발표대회 논문집 제17권 제1호 (2010. 4)

웹 서비스 환경에서 키 재발행을 위한 XKMS 기반 키 관리 기술에 대한 연구

김대현*, 이재성, 이경화, 신용태
숭실대학교 컴퓨터학과
{dhkim*, jslee, khlee}@cherry.ssu.ac.kr, shin@ssu.ac.kr

A Study on the XKMS-Based Key Management for Key Reissue in Web Services Environment

Dae-Hyun Kim*, Jae-Sung Lee, Kyoung-Hwa Lee, Yong-Tae Shin
Dept. of Computer, SoongSil University

요 약

최근 웹 기반의 XML이 인터넷 전자거래 및 데이터 전송에 이용되고 있다. 인터넷을 통해 비즈니스 거래가 신뢰성 있게 수행되기 위해서 XML 키 관리의 중요성이 높아지고 있다. XKMS는 XML 문서를 교환하기 위해 보안을 목적으로 사용되는 암호화키의 안전한 관리를 위한 명세이다[1]. 그러나 키의 분실 및 키 재발행 시 키의 변경에 따른 키 관리 방법에 대해서는 정의하지 않고 있다. 따라서 본 논문에서는 XKMS 표준 명세를 준수하는 키 변경에 따른 확장된 키 재발행(Extended-Reissue) 서비스를 제안한다. 제안하는 방식은 서버에 이전 사용자의 개인키를 저장하고, 키 분실 및 변경에 따른 키 재발행 시 분실키와 신규 키를 동시에 사용함으로써 효율적인 보안 서비스를 제공할 수 있다.

1. 서론

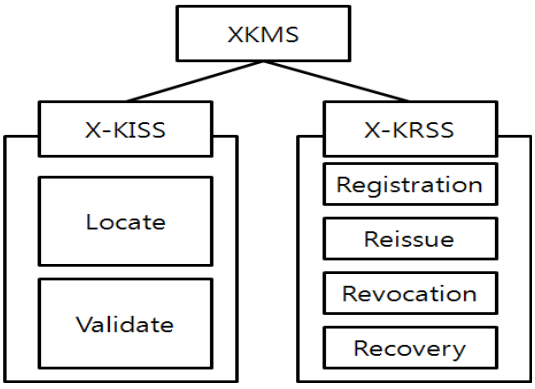
최근 인터넷의 급속한 발전으로 웹 서비스에 대한 중요성이 높아지고 있다. 또한 웹 기반의 XML(eXtensible Markup Language)이 인터넷 전자거래 및 데이터 전송에 이용되고 있다. 인터넷을 통해 비즈니스 거래가 신뢰성 있게 수행되기 위해서 XML 키 관리의 중요성이 높아지고 있다. OASIS(Oraganization for the Advancement of Structured Information Standards)와 W3C(World Wide Web Consortium)는 이러한 요구를 만족하기 위해 XML Encryption, XKMS(XML Key Management System)[1], SAML(Security Assertion Markup Language)[2], XACML(Extensible Access Control Markup Language)[3] 등과 같은 XML 기반의 표준 보안명세를 개발하고 있다.

암호화된 XML 문서 보안을 위해 키의 관리가 중요하다. XKMS는 XML 문서를 교환하기 위해 보안을 목적으로 사용되는 암호화키의 안전한 관리를 위해 등장했다. XKMS는 키 관리를 위해 XML 키 정보 서비스와 키 등록 서비스로 구성된다. 그러나 XKMS의 키 등록 서비스는 단순히 키의 등록 및 폐기와 관련된 내용만을 정의하고 키의 변경에 따른 키 관리 방법에 대해서는 정의하지 않고 있다. 따라서 본 논문에서는 XKMS에서 정의한 키 재발행 서비스를 강화한 확장된 키 재발행(Extended-Reissue) 서비스를 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 XKMS의 키 관리 구조 및 문제점을 살펴본다. 3장에서는 XKMS의 문제점을 개선하기 위한 확장된 키 재발행 서비스를 제안한다. 마지막 4장에서는 결론을 맺는다.

2. XKMS (XML Key Management specification)

XKMS[1]는 W3C(World Wide Web Consortium)에 의하여 주도적으로 표준화가 진행되고 있는 XML 키 관리 명세이다. XKMS는 암호 기능이 있는 XML 애플리케이션을 인증하기 위한 포괄적이고 개방적, 표준적인 접근방식을 취한다. XKMS는 X-KISS(XML Key Information Service Specification)와 X-KRSS(XML Key Registration Service Specification)의 두 영역으로 구성되어 있다. (그림 1)은 XKMS 서비스 구성도를 나타낸다.



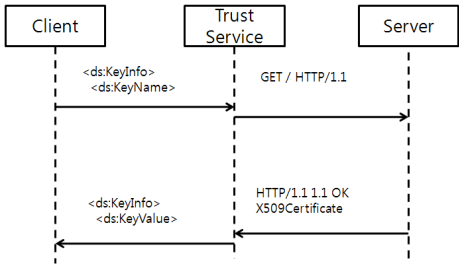
(그림 1) XKMS 서비스 구성도

2.1. XML 키 정보 서비스 (X-KISS)

X-KISS[1]는 XML 전자서명, XML 암호화된 데이터와 관련 키 정보 처리를 지원하기 위한 프로토콜로 PKI의 복잡성을 최소화하기 위해 설계되었다. X-KISS는 식별정보가 주어졌을 때, 송신자가 수신자에게 보낼 메시지를 암호화하기 위한 수신자의 공개키를 획득하는 키 위치 정보(Locate) 서비스와 획득한 공개키가 유효한지를 확인하기 위한 키 유효성 검사(Validate) 서비스로 구성된다.

2.1.1. 키 위치 정보 (Locate)

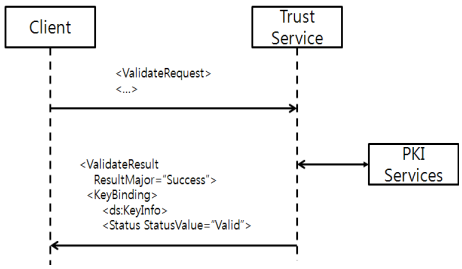
Locate Service[1]는 XML 서명을 포함하는 XML 문서를 검증해야 하는 애플리케이션이 있으면, 엘리먼트를 포함하는 XML 문서를 참조하는 파일이나 스트림, URI(Uniform Resource Identifier) 값을 받아들인다. 서명의 유효성을 검증하기 위해 클라이언트에서 인증 경로를 통해서 유효성을 확인한다. 송신자는 Trust Service를 통해 Server-A에 있는 수신자의 공개키를 획득한다. (그림 2)는 키 위치 정보 서비스를 나타낸다.



(그림 2) 키 위치 정보 서비스

2.1.2. 키 유효성 검사 (Validate)

Validate Service[1]는 키의 이름과 실제 공개키 간의 연결이 맞는지 확인하는 역할을 수행한다. 이것은 키 정보 서비스를 포함하고 있으므로 키의 이름, 공개키 간의 관계, 공개키의 위치를 알 수 있다. 송신자는 Trust Service를 통해 PKI(Public Key Infrastructure) Service에 접속하여 공개키를 획득하고 유효성 검사를 수행한다. (그림 3)는 키 유효성 검사 서비스를 나타낸다.



(그림 3) 키 유효성 검사 서비스

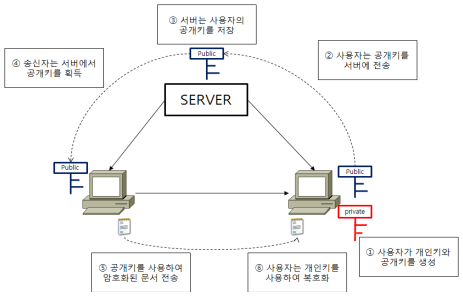
2.2. XML 키 등록 서비스 (X-KRSS)

X-KRSS[1]는 사용자가 XKMS와 관련하여 공개키와 개인키를 사용할 수 있도록 키의 등록을 지원하는 프로토콜이다[1]. 서버에 등록된 공개키는 수신자가 받을 문서 또는 데이터를 암호화하기 위해 사용되며, 수신자가 가지고 있는 개인키는 전송받은 데이터를 복호화하기 위해 사용된다. X-KRSS는 키 등록(Registration), 키 재발행(Reissue), 키 폐기(Revocation), 키 복구(Recovery) 서비스로 구성된다.

2.2.1. 키 등록 (Registration)

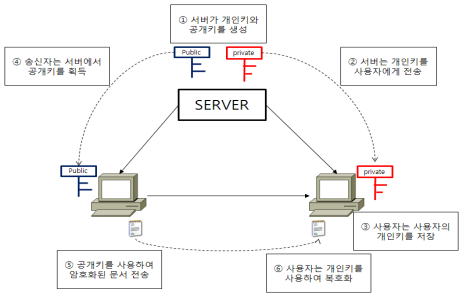
키 등록[1]은 키를 생성하는 주체에 따라 사용자에 의한 키 생성 방법과 서버에 의한 키 생성 방법으로 나뉜다.

사용자에 의한 키 생성 방법은 사용자가 공개키와 개인키를 생성한 후, 공개키는 서버에 전송하고, 개인키는 사용자가 보관한다. 이 방법에서는 개인키가 통신을 통해 전송되지 않고 사용자가 보관하기 때문에 높은 보안성을 유지할 수 있다. 그러나 등록된 개인키가 분실될 경우, 서버는 사용자의 개인키를 알지 못하기 때문에 분실된 개인키를 획득할 수 없다는 단점이 있다. (그림 4)는 사용자에 의한 키 생성 방법을 나타낸다.



(그림 4) 사용자에 의한 키 생성 방법

서버에 의한 키 생성 방법은 서버에서 사용자의 공개키, 개인키 쌍을 생성하고 사용자에게 안전한 통신방법으로 개인키를 전송해 준다. 이 방법에서는 개인키가 서버에 저장되기 때문에, 사용자가 개인키를 분실 하였을 경우 개인키 복구 및 획득이 용이하다. (그림 5)는 서버에 의한 키 생성방법을 나타낸다.



(그림 5) 서버에 의한 키 생성 방법

2.2.2. 키 재발행(Reissue)

키 재발행[1]은 이전에 발행한 키를 사용자에게 재발행하는 것이다. 사용자의 개인키가 만료되거나 분실된 경우, 또는 키 폐기가 일어난 경우 이 과정을 통해 키가 재발행된다. 키 재발행을 위한 요청 과정은 초기 키 등록 과정과 유사하다.

2.2.3. 키 폐기 (Revocation)

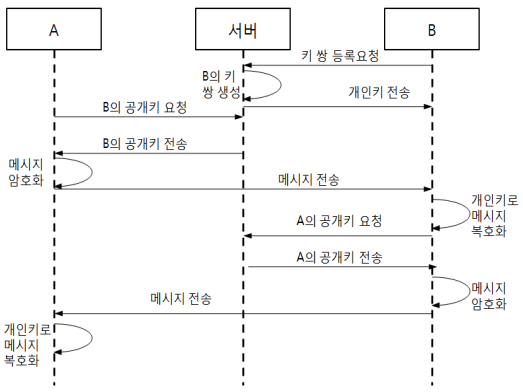
키 폐기[1]는 클라이언트가 이전에 요청했던 키 등록 정보를 폐기하는 것이다. 이 서비스는 키 폐기 요청을 전송함으로써 시작된다. 서버는 요청자의 인증을 수행하고, 서버에 저장되어 있는 사용자의 키 쌍을 폐기한다. 서버에 키 정보가 없으면 서버는 요청자에게 키 정보가 없음을 전달한다.

2.2.4. 키 복구 (Recovery)

키 복구[1]는 이전에 발행한 키가 서버에 저장되어 있는 경우 이 키를 사용자에게 재발행하는 것이다. 키 복구는 요청자의 인증을 수행하고 사용자의 개인키가 서버에 저장 되었는지 확인한다. 서버에 키 정보가 없으면 서버는 요청자에게 키 정보가 없음을 전달한다. 키 재발행을 위한 요청은 서버에서 이전의 키를 검색하는 과정을 제외하고는 초기 키 등록 과정과 유사하다.

2.3. XML기반 키 생성 및 메시지 송수신 절차

(그림 6)은 초기 서버에서 공개키, 개인키 쌍을 생성하고, 생성된 키를 이용해 메시지를 암복호화 하는 절차를 나타낸다.



(그림 6) 키 생성 및 메시지 송수신 절차

① B는 서버에 키 쌍 등록을 요청한다.
② 서버는 생성된 키 쌍을 B에게 전송한다.
③ A는 서버에게 B의 공개키를 요청한다.
④ 서버는 A에게 B의 공개키를 전송한다.
⑤ A는 B의 공개키로 암호화된 메시지를 B에게 전송한다.
⑥ B는 전송받은 메시지를 자신의 개인키로 복호화한다.
⑦ B는 서버에게 A의 공개키를 요청한다.
⑧ 서버는 B에게 A의 공개키를 전송한다.
⑨ B는 A의 공개키로 암호화된 메시지를 A에게 전송한다.
⑩ A는 전송받은 메시지를 자신의 개인키로 복호화한다.

2.4. 문제점

본 절에서는 XKMS에서 정의한 XML 키 정보 서비스와 키 등록 서비스에 대해 분석하였다. XKMS의 키 등록 서비스는 단순히 키의 등록 및 폐기와 관련된 내용만을 정의하고 있다. 즉, 키의 분실 및 키 재발행 시 키의 변경에 따른 키 관리 방법에 대해서는 정의하지 않고 있다. 따라서 본 논문에서는 XKMS 표준 명세를 준수하는 키 변경에 따른 확장된 키 재발행(Extended-Reissue) 서비스를 제안하고자 한다.

3. 제안하는 키 재발행(Extended-Reissue) 서비스

3.1. 서비스 모델

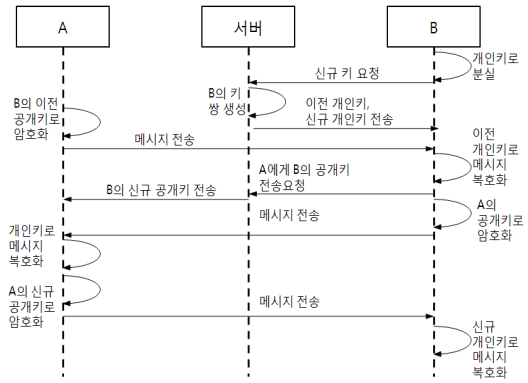
우리는 개인키의 분실 또는 해킹으로 인하여 키의 변경이 필요한 경우가 있다. 이때 보안에 사용하던 키를 서버에 보관하지 않을 경우, 분실된 키를 사용해 암호화된 메시지를 복호화 할 수 없다. 따라서 안정적인 복호화를 보장하기 위해서 우리는 이전 키의 보관이 필요하다.

사례 1 : 사용자 A는 매월 인터넷으로 고지서를 전달 받는다. 수신된 고지서는 사용자 A의 공개키로 암호화되어 전송된다. 사용자 A는 자신의 개인키로 해당 메시지를 복호화한 고지서를 확인한다. 송신자는 고지서를 암호화하기 위한 사용자 A의 공개키를 초기에 한번 발급받고, 이후 사용자 A로부터 변경 요청이 있기 전까지는 해당키를 사용한다.

사례 2 : 사용자 B는 자신이 사용하던 개인키를 분실해 서버에 키 재발행 요청을 하였다. 서버는 사용자 B의 공개키, 개인키 쌍을 새로 만들고 사용자 B의 이전 개인키와 신규 개인키를 사용자 B에게 전송한다. 그리고 서버는 사용자 B의 이전 공개키를 사용하는 사용자들에게 사용자 B의 신규 공개키를 전송한다.

3.2. 확장된 키 재발행(Extended-Reissue) 절차

(그림 7)은 개인키 획득을 위한 확장된 키 재발행 절차를 나타낸다.



(그림 7) 확장된 키 재발행 절차

① B는 자신의 개인키가 분실된 것을 인지한다.
② B는 서버에 신규 키 재발행을 요청한다.
③ 서버는 신규 키 쌍을 생성하고, B의 이전 개인키와 신규 개인키를 사용자에게 전송한다.
④ A는 이전에 전송받은 B의 공개키로 암호화된 메시지를 B에게 전송한다.
⑤ B는 이전 개인키로 메시지를 복호화한다.
⑥ B는 서버에 A에게 자신의 신규 공개키를 전송할 것을 요청한다.
⑦ 서버는 A에게 B의 신규 공개키를 전송한다.
⑧ B는 이전에 전송받은 A의 공개키로 암호화된 메시지를 A에게 전송한다.
⑨ A는 전송받은 메시지를 자신의 개인키로 복호화한다.
⑩ A는 전송받은 B의 신규 공개키로 암호화된 메시지를 B에게 전송한다.
⑪ B는 전송받은 메시지를 자신의 신규 개인키로 복호화한다.

확장된 키 재발행 절차는 일반적인 키 재발행 절차와 달리 이전에 메시지를 보낸 사람에게 메시지를 보낼 때 서버에 공개키를 요청하는 작업을 생략하였고, 사용자가 자신의 이전 개인키를 저장하는 작업을 추가 하였다. 이러한 과정은 수신자가 이전 공개키로 암호화된 메시지를 복호화할 수 있고, 송신자가 메시지를 보내기 전 수신자의 공개키를 획득하는 절차를 생략함으로써, 프로세스 간소화에 좋다.

4. 결론

본 논문에서는 XKMS에서 정의한 XML 키 정보 서비스와 키 등록 서비스에 대해 분석하였다. 또한 분석된 내용을 바탕으로 안전한 XML 웹서비스를 위한 키 변경에 따른 확장된 키 재발행(Extended-Reissue) 서비스를 제안하였다. 제안하는 방식은 서버에 이전 사용자의 비밀키를 저장하고, 키 분실 및 변경에 따른 키 재발행 시 분실키와 신규 키를 동시에 사용함으로써 효율적인 보안 서비스를 제공할 수 있다.

향후 제안하는 키 재발행 서비스에서 사용자에게 개인키를 보다 안전하게 전송할 수 있는 기술의 연구가 필요하다. 또한 XKMS는 현재 PKI 시스템과의 연동을 전제로 설계되어 있다. 이러한 시스템을 제공하기 위해서는 보다 효율적인 기존 시스템과의 연동에 관한 많은 연구가 선행되어야겠다.

참고문헌

[1] XML Key Management WG, “XML Key Management Specification (XKMS 2.0),” W3C, 28 Jun. 2005, http://www.w3.org/TR/2005/REC-xkms2-20050628
[2] OASIS SS TC, "Security Assertion Markup Language(SAML) ver.2.0 Technical Overview," OASIS, 20 Feb. 2005, http://xml.coverpages.org/SAML-TechOverview20v03-11511.pdf
[3] OASIS XACML TC, "eXtensible Access Control Markup Language (XACML) ver.2.0," OASIS, 30 Sep. 2004, http://xml.coverpages.org/XACMLv20CD-CoreSpec.pdf
[4] XML Signature WG, "XML Signature Syntax and Processing (Second Edition)," W3C, 10 Jun. 2008, http://www.w3.org/TR/2008/REC-xmldsig-core-20080610
[5] XML Encryption WG, "Decryption Transform for XML Signature," W3C, 10 Dec. 2002, http://www.w3.org/TR/2002/REC-xmlenc-decrypt-20021210