XML을 이용한 그룹 키 관리 기법 설계

이영경°, 장성렬", 이경현'
부경대학교 전자계산학과'
부경대학교 정보보호학과"

{twinkle°, jiya686, rhee}@lisia21.net

Design of group key management scheme using the XML

Youngkyung Lee°' Sungryul Chang" Kyunghyune Rhee
Dept. of Computer Science, PuKyong Nat'l University'
Dept. of Information Security, PuKyong Nat'l University"

요 약

그룹 키는 어플리케이션 계층에서 많이 사용하고 있으며, 이는 각자 다른 환경에서 동작하는 사례가 많다. 따라서, 이기종간의 환경에서 동작하는 그룹 키를 언어 중립적이고 플랫폼 독립적인 XML을 사용하여 통합가능한 환경의 구성을 위해 그룹 키 프로토콜을 설계하고자 한다. 이는, XML의 특성상 어떠한 환경에서도 영향을 받지 않으므로 어느 플랫폼이나 적용이 용이한 이점을 가진다.

1. 서 론

인터넷상에서 전자상거래나 XML 전자서명, XML 암호, 원격회의 등이 지원됨에 따라 물리적인 활동량이 감소했을 뿐만 아니라 생활이 편리하게 되었다. 이러한 인터넷상에서 신뢰성 있는 거래를 가능하게 해 주는 주요한 요소 중의 하나가 바로 안전한 키의 관리이다. 본 논문에서는, 키 관리 부분 중에서도 특히 분산형 환경에서 동작하는 그룹 키 관리를 다루고자 한다. 각 응용계층에서 유지되는 그룹은 크게 구성원의 규모나 구성원의 탈퇴, 가입여부, 통신방향 등에 따라 다양한 형태를 가지고 있으며, 그에 따라 지원해야 하는 그룹 키의 갱신방법과 관리 기능이 다르게 요구된다. 이렇게 각각 다른 환경에서 동작하는 그룹관리를 XML을 이용하여 구현함으로써 모든 플랫폼에서 적용할 수 있도록 한다. XML은 2003년 UN이 공식 표준 언어로 승인함으로써 XML의 활용도는 갈수록 높아지고 있으며, 웹에서는 없어서는 안될 인터넷 표준언어로 자리매김 하고 있다. 본 연구에서는 TGDH 방식을 기본으로 한 그룹 키 관리 프로토콜을 XML 스키마로 설계하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 그룹 키에 대한 연구를 살펴보고, 3장에서는 본 논문에서 제시하고자 하는 프로토콜의 특징을 알아본다. 또한, 스키마를 설계하고 4장에서는 결론을 맺는다.

2. 그룹키

◆ 표기법

- M_i : i번째 그룹 멤버
- BK_i : 멤버 M_i의 블라인드 키
- BK_i^* : 멤버 M_i의 블라인드 키들의 집합
- T_i : 멤버 이벤트가 발생한 후 수정된 트리
- $<l,v>$: 트리에서 l 레벨, v번째 노드
- p : 큰 소수
- α : exponentiation base

TGDH(Tree based Groupkey Diffie-Hellman)방식[1][4]은 분산형 방식으로써 멤버의 참여나 탈퇴 시 키 갱신을 수행할 경우, 그룹내의 모든 멤버가 그룹 키를 만드는데 참여하는 방식을 말한다. 이 방식은 키 관리 서버가 필요하지 않으므로 중앙집중형 방식에서 일어나는 키 관리 서버가 공격자로부터 공격당했을 경우, 그룹내의 모든 키가 노출되는 위험에서 벗어날 수 있다[3]. 또한, 이 방식은 키 트리에서 새로운 그룹 키를 계산하기 위해 일방향 함수를 키 트리에 상향식으로 적용하였다. 일반적인 키 트리 기법처럼 사용자의 비밀키를 단말 노드로 가지고 형제(sibling) 노드의 블라인드 키(Blind Key)와 함께 한 단계 상위의 부모노드의 비밀키를 만들어낸다. 이 키가 intermediate key가 되며, 계층적으로 암호학적 연산이 이루어진다. 이때, 루트노드가 그룹멤버들이 공동으로 소유하는 그룹 키가 되며 중간 노드의 키들이 이진 키 트리를 구성한다. 키 트리에서 노드 M은 노드 키 $K_{<l+1,2v>}$와 함께 형제 노드의 블라인드 노드 키라는 $BK_{<l+1,2v+1>}$ 두 개의 키와 연관되어진다. 이때, 블라인드 노드키는 $BK_{<l,v>}=f(K_{<l,v>})$ 이며, 함수 $f(k)=\alpha^k \bmod p$ 이다. 즉, 블라인드 키는 실제 키의 값을 직접 드러내지 않도록 숨기는 역할을 한다. $K_{<l,v>}=BK_{<l+1,2v+1>}^{K_{<l+1,2v>}} \bmod p$ 로 계산할 수 있다. 키 트리의 구조에서 중간 노드들은 모두 두 개의 자

식 노드를 가지며 단말노드는 각 멤버 자신이 선택한 비밀키는 각 멤버의 비밀키로 할당한다.

3. 제안하는 XML 스키마 설계

본 논문에서 제안하고자 하는 그룹 키 관리 기법은 모바일이나 차세대 전자상거래의 표준인 XML에 기반하고 있으므로 이기종 시스템간에 접목할 수 있다. 그룹 키는 항상 유동적으로 작동하며, 멤버의 참가(join)와 탈퇴(leave), 합병(merge), 분할(partition) 등의 이벤트가 발생한다.

본래 TGDH는 sponsor를 가지는데 이 sponsor의 기능은 이벤트가 발생한 경우 각 멤버들에게 그 사실을 알리고 그룹 키 갱신을 하는데 참여하는 역할을 한다. 본 논문에서는 sponsor가 DOM을 이용하여 키 분배에 해당되는 모든 메시지를 XML 형태로 넘겨주는 역할을 한다. 하단에서는 그룹 키에서 멤버의 참가(join)와 탈퇴(leave) 등의 이벤트가 일어났을 경우를 두 가지로 나누어 보여주고 있다.

①새로운 멤버 M_{n+1}가 그룹에 참가하고자 할 경우

멤버 M_{n+1}는 모든 멤버들에게 참가 요청과 함께 자신의 블라인드 키 값 BK_{n+1}을 모든 멤버에게 전송한다. 그림1은 새로운 멤버 M_4가 참가한 경우 트리 업데이트를 보여주고 있다.

M_{n+1} ------→ $Member\ \{M_1,\cdots,M_n\}$

Join request

BK_{n+1}

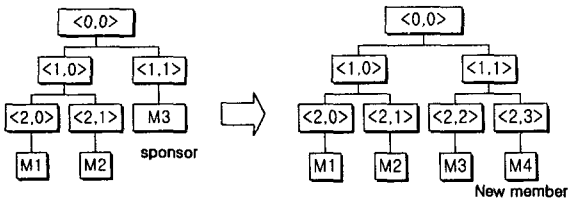

그림 1 멤버 join시 트리 구조

```
<xsd:element name="JoinRequest"/>
<xsd:complexType>
    <xsd:sequence>
        <xsd:element name="Message" type="xsd:String"/>
    </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="JoinResponse"/>
<xsd:complexType>
    <xsd:sequence>
        <xsd:element name="Message" type="xsd:String"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType>
    <xsd:sequence>
        <xsd:element name="PublicValue"type="xsd:CryptoBinary"/>
    </xsd:sequence>
    <xsd:attribute name="UserId" type="xsd:ID"/>
</xsd:complexType>
```

표1 멤버참가 시 요청(상), 응답(하) 메세지 구조

표1은 sponsor에게 참가 요청 메시지를 보내고 그에 대한 응답으로 ID를 부여받는 XML 스키마 구조를 보여주고 있다. 모든 멤버는 참가되는 새로운 멤버에 의해 키 트리를 새로운 노드로 업데이트하고 sponsor와 연관된 leaf 노드로부터 모든 키와 BK를 삭제한다.

$Member\ \{M_1,\cdots,M_{n+1}\}$ ←------ $Sponsor$

$T_s(BK_s^*)$

표2는 멤버참가 시 갱신되는 블라인드 키들의 집합과 트리 구조를 XML 스키마 구조로 나타낸 것이다[6]. <AddNodeLocate>엘리먼트는 멤버참가 시 업데이트 되는 트리의 위치정보를 나타내고, <BlindKeyValue>엘리먼트는 멤버참가 시 업데이트 되는 키 값들을 나타낸다. 그림2에서 갱신되는 블라인드 키들의 집합은 {<2,3>,<1,1>}이 된다. 이들에 대한 트리 위치정보와 블라인드 키들을 XML 문서에 담아서 멤버들에게 전송하며, 블라인드 키는 속성의 ID를 통해서 구분한다.

```
<xsd:element name="BroadcastContents"/>
<xsd:complexType>
  <xsd:sequence>
    <xsd:element name="AddNodeLocate"
                 minOccurs="1" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="depth" type="xsd:integer"/>
          <xsd:element name="length" type="xsd:integer"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType>
  <xsd:sequence>
    <xsd:element name="BlindKeyValue"
                 minOccurs="1" maxOccurs="unbounded" >
      <xsd:complexType>
        <xsd:sequence minOccurs="0">
          <xsd:element name="PublicValue"type="xsd:CryptoBinary"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="UserId" type="xsd:ID"/>
</xsd:complexType>
```

표2 멤버 참가시 XML 스키마 구조

② 임의의 멤버 M_l이 탈퇴하는 경우

탈퇴하고자 하는 M_l는 모든 멤버에게 자신의 탈퇴사실을 알린다.

M_l ------→ $Member\ \{M_1,\cdots,M_n\}$

Leave request

모든 멤버는 탈퇴 멤버와 그와 관련된 부모노드를 삭제하고, sponsor와 연관된 leaf 노드로부터 모든 키와 BK들의 집합을 삭제한다. Sponsor는 새로운 share를 생성하고 탈퇴 멤버만 제외한 모든 멤버들에게 블라인드 키들의 집합 BK_s^*를 포함한 트리 T_s를 멤버에게 전송한다.

```
<xsd:element name="LeaveRequest"/>
<xsd:complexType>
    <xsd:sequence>
        <xsd:element name="Message" type="xsd:String"/>
    </xsd:sequence>
</xsd:complexType>
```

표 3 멤버삭제 시 요청 메시지

T_s는 갱신된 트리정보를 담고 있는 것으로써 표4에 <RemoveNodeLocate>엘리먼트를 통해 표현하고 있다. 그림2는 멤버 탈퇴 시 갱신되는 트리를 그림으로 표현한 것이다.

$\{M_1,\cdots,M_{n+1}\}-\{M_l\} \leftarrow\text{------}$ *Sponsor*

$T_s(BK_s^*)$

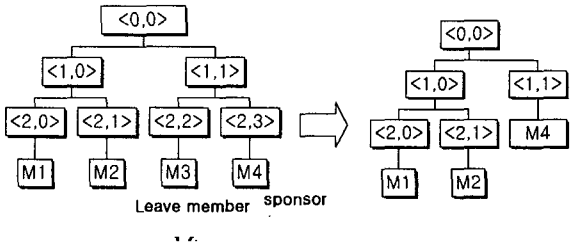

그림 2 멤버 leave 시 트리 구조

```
<xsd:element name="BroadcastContents"/>
<xsd:complexType>
  <xsd:sequence>
    <xsd:element name="RemoveNodeLocate"
                 minOccurs="1" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="depth" type="xsd:integer"/>
          <xsd:element name="length" type="xsd:integer"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType>
  <xsd:sequence>
    <xsd:element name="BlindKeyValue"
                 minOccurs="1" maxOccurs="unbounded">
      <xsd:complexType>
       <xsd:sequence minOccurs="0">
        <xsd:element name="PublicValue" type="xsd:CryptoBinary"/>
       </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="UserId" type="xsd:ID"/>
</xsd:complexType>
```

표 4 멤버 삭제시 XML 스키마 구조

표4는 멤버 삭제시 XML 스키마 구조를 나타내고 있다. 멤버 삭제시 갱신되는 블라인드 키들의 집합 BK_s^*는 {<1,1>}이 된다.

Sponsor는 <RemoveNodeLocate>엘리먼트에 키 트리 정보를 담고, <BlindKeyValue>엘리먼트에 갱신되는 키 정보를 담아서 멤버들에게 전송한다.

4. 결론 및 향후 연구방향

XML은 어떤 형의 데이터라도 구조적인 방법으로 데이터를 기술할 수 있는 마크업 언어를 생성하기 위한 언어로 어플리케이션에서 쉽게 텍스트로 파싱(parsing)할 수 있는 장점을 가지고 있다[9]. 이러한 장점을 이용하여 본 논문에서는 XML을 기반으로 한 그룹 키 프로토콜을 설계하였다. XML의 특성상 그룹 멤버들의 서로 다른 이기종간의 환경에서도 영향을 받지 않고, XML로 형태로 그룹키를 교환함으로써 각각의 멤버들이 어느 플랫폼에서나 동작이 가능하도록 설계되었다.

논문에서는 언급되지 않았지만 그룹 키에서 키 분배, 교환은 PKI 기반을 전제하고 수행된다. 따라서, 본 논문을 확장하여 XML 키 관리 명세를 다루는 XKMS를 이용하여 그룹 키 서명, 암호화를 효율적으로 도와줄 수 있는 XKMS(XML Key Managment Specification)[2]서버를 향후 과제로 구현할 계획이다. 이는, XML을 기반으로 메시지를 주고 받으므로 모바일 통신 시 그룹을 형성해서 통신하고자 할 때, XKMS 서버를 이용하면 보다 효율적인 그룹통신을 위한 키 관리 기법이 될 것으로 판단된다.

5. 참고 문헌

[1]Yongdae Kim, Adrian Perrig, Gene Tsudik, "Tree-based Group Key Agreement", 2001

[2]Draft version 1.1 draft 4, XML Key Management Specification(XKMS), 2001.

[3]Ghassan Chaddoud, Isabelle Chrisment, Andre Schaff, "Dynamic Group Communication Security", IEEE, 2003

[4]Y.Kim, A. Perrig, G. Tsudik, "Simple and fault-tolerant key agreement for dynamic collaborative groups", Proceedings of th 7th ACM conference on Computer and communications security, 2000

[5]W3C, XML Encryption Syntax and Processing, http://www.w3.org/TR/2002/CR-xmlenc-core-20020802/, 2002.

[6]W3C, "http://www.w3.org/TR/xmlschema-2/", 2002

[7]조현호, 박영호, 이경현, "OFT를 사용한 안전한 멀티캐스트 프로토콜", 한국정보처리학회발표논문집, 2001

[8]박영호, 이경현, "이동 네트워크 환경에서의 그룹 키 관리구조", 한국정보보호학회 논문지, 2002.

[9]Blake, XML Security, Mc Graw Hill, 2002