제26회 한국정보처리학회 추계학술발표대회 논문집 제13권 제2호 (2006. 11)

추가, 삭제 연산을 위한 XML 접근제어의 확장

이진형*, 이은정**
경기대학교 전자계산학과
*polyam@gmail.com **ejlee@kyonggi.ac.kr

Extension of XML Access Control for Insert and Delete Operation

Jinhyoung Lee*, Eunjung Lee**
*Computer Science, Kyonggi University

요 약

XML문서에 대한 보안요구가 커지면서 XML문서의 통신상 보안과 관리적 측면에서의 보안에 관련된 연구가 계속되고 있다. 이중 관리적 측면에서의 보안은 접근제어를 통해 제공할 수 있다. XML 문서에 대한 접근제어와 관련된 연구들은 XML 문서에 대한 읽기 연산을 중심으로 연구되어왔다. 그러나 XML에 대한 연산은 읽기뿐만 아니라 추가, 삭제연산도 고려해야한다. 따라서 접근제어에서도 추가, 삭제 연산에 대한 연구가 필요하다. 본 논문에서는 추가, 삭제 읽기 연산간의 계층관계를 설정하고 이를 이용한 추가, 삭제 연산을 포함한 역할기반 XML 권한기술 모델을 제안하였다. 또한 권한기술 모델을 이용한 적용사례를 웹 포탈사이트 시나리오를 통해 제시하였다.

1. 서론

W3C에서는 XML을 웹 데이터의 표준으로 제정하였고[7] 그 이후 많은 데이터들이 XML로 표현되기 시작했다. 현재는 다양한 기관, 사용자들이 XML 형식의 문서를 사용하고 있다. 이러한 문서들이 웹상에서 XML 문서는 네트워크를 통해 전달되고 공유되어 문서의 기밀성을 유지하는 것이 요구된다. 때문에 XML 문서보안에 대한 연구들은 주로 암호화, 전자서명, 키관리 등과 같은 통신상의 보안을 중심으로 이루어지고 있다. 하지만 XML 문서가 거대해지고 복잡해짐에 따라 통신상의 보안뿐만 아니라 관리적인 측면에서의 보안이 필요하게 되었다. 관리적인 측면에서의 보안은 접근제어를 통해 보장 할 수 있다. 접근제어를 통하여 자원 혹은 접근 요청 개체들에 권한 부여를 통해 자원에 대한 접근제어를 수행함으로써 관리적 측면에서의 보안기능을 제공할 수 있다. 접근제어를 위한 권한은 사용자 개별로 부여하지 않고 역할을 기반으로 권한설정을 용이하게 할 수 있다.[1]

하지만 이전의 연구들은 대부분이 추가, 삭제 연산에 대한 고려 없이 읽기 연산을 중심으로 연구되어왔다. 따라서 본 논문에서는 XML 접근제어에서의 추가, 삭제 연산에 대하여 계층관계를 설정하고 이를 이용한 접근제어를 위한 권한기술 모델을 제안하고자 한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 관련연구로 기존의 접근제어와 관련한 연구들을 살펴보고, 3장에서는 제안하는 모델에 대하여 살펴보고, 4장에서는 제안하는 모델을 적용한 시나리오를 제시하고, 5장에서는 결론 및 추후연구에 대해 기술한다.

2. 관련연구

2.1 역할기반 접근제어 모델

Sandhu등이 제안한 역할기반 접근제어 모델에서는 역할을 기반으로 권한을 설정하고 역할을 사용자에게 부여함으로써 권한의 관리 편의성을 제공하고자 하였다. Sandhu 등은 User, Role, Permission,

Session을 기본으로 한 RBAC0를 제안하고, Role의 상속을 포함한 RBAC1, 제약사항(Constraint)을 추가한 RBAC2, RBAC1과 RBAC2를 결합한 RBAC3를 역할기반 접근제어 모델로 제시하였다.[6]

2.2 XML 접근제어

Sandhu등이 제안한 역할기반 접근제어 모델이 갖는 권한관리의 편의성 때문에 역할기반의 XML 접근제어 방법에 대한 연구가 있었다.[4][5] 또한 XML 형식의 문서는 다른 영역에서의 접근제어와 달리 문서 자체가 사용자 정의 구조를 가지고 있는 특성에 따라 접근제어의 대상을 기술하는 다른 방법이 필요하다.

Irini Fundulaki 등은 XML 접근제어에서 접근대상을 기술하는 방법으로 XPath와 XPath Filter를 이용한 방법을 제시하였다.[3]

XML 문서에 대한 접근제어 방법으로 Jason Crampton은[4] 역할기반의 XML 접근제어 방법을 제시하였는데 특히 역할(Role)의 상속을 적용하였다. 제어대상이 되는 서브트리에 대하여 각각의 키를 만드는 것이 아니라 역할의 상속을 통해 키를 상속하고 역할별로 키를 만들어 역할당 하나의 키를 가짐으로써 한 역할이 여러 개의 키를 갖는 복잡성을 제거하고자 하였다.

또 다른 XML 데이터에 대한 접근제어 방식으로 Function기반의 접근제어 방법이 있는데 Naizhen Qi 등은 접근제어를 수행하는 방법으로 Rule-Function을 제안하였다.[6] 접근제어 정책에 따라 각각의 Rule-Function을 생성하여 Rule-Function의 반환값에 따라 접근여부를 판단하여 접근제어를 수행하였다.

대부분의 이전 연구가 읽기 연산을 위주로 연구되었다. 임정환 등은 업데이트 연산을 고려한 XML 문서 접근제어를 제안하였다.[2] 업데이트 연산의 위해 업데이트 연산을 위한 업데이트 Operator를 재정의 하고 Action Type을 정의하였다. 특히 삽입, 삭제와 같은 업데이트 연산을 문서의 구조를 바꾸는 경우와 그렇지 않은 경우를 구분하는데 그쳤다. 따라서 본 논문에서는 업데이트 연산에 의해 구조가 변경되지 않는 경우를 고려하여 추가, 삭제 연산사이의 연관성을 정의였고 이를 이용한 권한기술 모델에 대하여 연구하였으며 권한기술 모델을 이용한 시스템 시나리오를 제시하였다.

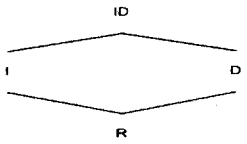
3. 제안 모델

3.1 Action Type

XML문서에 대한 연산으로는 추가, 삭제, 읽기, 수정 등의 연산이 있다. 읽기 연산에 대해서는 모든 노드가 가능하고 삭제는 문서의 유효성을 유지하기 위해 반복부 노드, 추가는 추가하고자 하는 노드의 부모노드에서 발생한다.

제안하는 모델에서는 추가, 삭제, 읽기 연산 간의 상관성에 따라 연산의 집합을 구성하고, 이러한 연산의 집합을 ActionType으로 정의하였다. 본 논문에서는 읽기 권한이 있는 경우에만 추가, 삭제가 가능하다고 보고 연산의 집합을 정의하였다.

연산의 집합에 따라 읽기 연산만 가능한 R Type, 추가, 읽기 연산이 가능한 I type, 삭제, 읽기 연산이 가능한 D type, 추가, 삭제, 읽기 연산이 모두 가능한 ID type의 4가지로 정의하였다. 이들 ActionType들을 권한의 측면에서 보았을 때 ActionType간에는 (그림 1)과 같은 계층관계가 성립한다.



(그림 1) ActionType의 계층관계

3.2 object

기존의 접근제어 모델에서와 같이 접근제어의 대상이 되는 노드를 object라 한다. object의 표현 XPath를 이용하며 특정 조건을 만족하는 대상을 선택하기 위하여 XPath 조건식을 사용한다. 사용자의 권한 표현시 사용자의 ID같은 가변적인 값을 표현하기 위하여 변수를 이용한다. XPath 조건식에서 변수에는 '$'붙여 구분한다.

본 연구에서는 추가, 삭제가 가능한 대상을 문서의 유효성을 유지할 수 있도록 추가, 삭제 연산 대상을 반복부로 한정하였다. Action Type이 I, ID인 경우 object에 표현된 노드는 실제로 추가 연산이 가능한 노드를 의미한다. 예를 들어 Action Type이 I Type 이고 object가 root/cafes/cafe인 경우 cafe 노드를 cafes 노드의 자식 노드로서 추가 할 수 있다는 의미이다. 같은 object에 Action Type이 D Type인 경우도 마찬가지로 cafes노드의 자식 노드

인 cafe를 삭제할 수 있다는 의미를 가진다.

3.3 권한 요소 표현

본 논문에서는 권한 요소의 표현으로 <subject, action, object, propagation>의 형식으로 기술한다. 제안하는 모델에서 사용하는 요소 중 subject는 사용자 또는 접근대상을 의미하는 다른 접근제어 방법과 동일한 의미를 갖는다. 예를 들어 "카페 운영의 권한은 본인이 소유한 카페의 회원노드를 읽고, 삭제할 수 있다"라는 내용에 대한 권한기술은 다음과 같이 할 수 있다.

<CafeManager, D, root/Cafes/Cafe[@Owner = $UserID]/Members/Member, L>

이와 같이 기술된 경우 카페 운영자의 역할을 가진 UserID에 따라 변수값을 결정하게 되고 결정된 값에 의해 Member라는 노드에 대한 권한이 주어지게 된다.

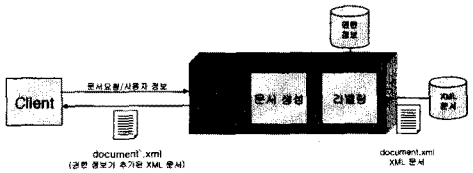
Action Type이 추가 또는 삭제를 포함한 Action Type인 경우 object는 앞서 가정함과 같이 추가, 삭제 연산에 의해 문서구조가 변경되지 않도록 propagation이 L인 경우는 object에 기술된 노드가 반복부여야만 하고, R인 경우는 object에 기술된 노드 또는 그 하위노드에 반복부가 하나 이상 포함되어 있어야만 한다. 그리고 propagation이 R이고 object에 기술된 노드와 그 하위노드가 반복부와 반복부가 아닌 노드가 혼재할 경우는 추가, 삭제는 반복부에서만 발생할 수 있으며 반복부가 아닌 노드의 경우는 읽기 연산만 가능하다.

3.4 권한 정책의 표현

권한 정책은 위에서와 같은 권한 요소와 역할 배정을 포함한다. 권한 배정은 사용자에게 역할을 할당하며 <User, Role>의 형식으로 기술하여 User에 세 Role을 배정한다. 권한 배정에서 <Role, Role>의 형식의 기술도 가능하며 이때는 역할간의 포함관계를 의미한다.

4. 적용 시나리오

제안된 모델을 사용한 시스템은 사용자가 요청한 문서를 제공하는 Server와 요청한 문서를 사용자에게 제공하는 Client로 구분된다. Server는 사용자의 역할과 기술되어있는 권한 정책을 분석하여 접근권한 정보가 추가된 문서를 전송한다. Client는 Server에서 전송된 문서를 분석하여 사용자에 권한에 허용된 형태의 View를 제공한다.

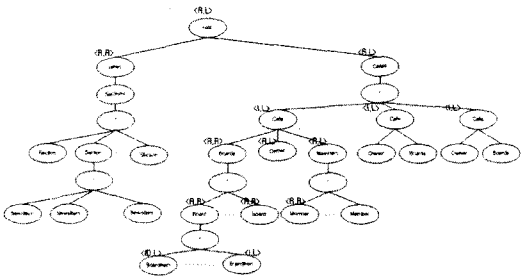


(그림 2) 시스템 구성

4.1 권한정보 계산

Server에서는 권한정보를 분석하여 사용자의 역할에 따라 보낼 권한정보가 추가된 문서를 생성한다. 이때 노드에 대한 추가되는 권한 정보는 Action Type과 Propagation으로 표현된다. 예를 들어 CafeMaster의 역할을 부여 받은 사용자에게는 <CafeMaster, D, root/Cafes/ Cafe, ,L>로 기술된 권한부분에 의해 Cafe 노드에 <D, L>라는 라벨을 추가한다.

동일한 노드에 적용가능한 권한요소가 여러 개인 경우 권한 부여를 위한 라벨링을 연구를 진행 중이다.



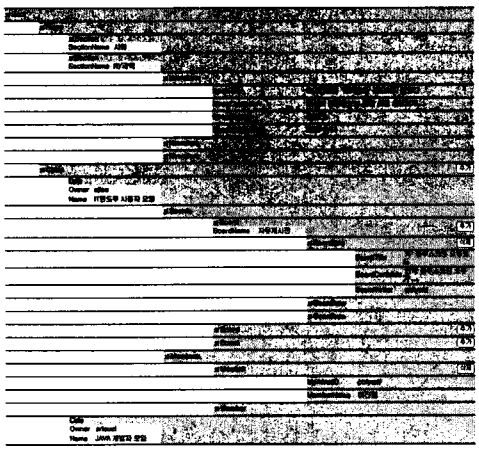
(그림 3) 라벨링된 Tree

4.3 Client의 권한정보 활용

Client는 서버로부터 권한 정보가 추가된 문서를 수신하게 된다. 수신한 문서에 추가된 권한 정보를 기반으로 사용자에게 View를 생성하여 준다.

Client는 각 노드에 추가된 ActionType를 분석하여 해당노드의 가시여부 및 추가, 삭제 가능여부에 따라 사용자의 View를 생성하여 준다.

이때 ActionType의 속성값이 ID, I인 경우 삽입 버튼은 부모노드에 표시되고 삭제 가능한 노드 즉 ID, D인 경우는 해당노드의 옆에 삭제 버튼이 위치하게 된다.



(그림 4) Client View

(그림 4)는 (그림 3)과 같은 라벨링에 의해 권한 정보가 추가된 문서를 Client에서 View를 생성한 것이다. 여기서 Root/Cafes/Cafe, Root/Cafes/Cafe /Boards/Board/BoardItem의 추가가 가능하도록 권한이 설정되어 있으므로 View에서 그들의 부모노드에 추가 버튼이 생성되었음을 볼 수 있다. 또한 삭제 권한이 있는 노드들에 대해서는 해당노드의 옆에 삭제 버튼이 추가되었음을 확인할 수 있다.

5. 결론

본 논문에서는 추가, 삭제를 고려한 권한 기술 모델을 제안하였다. 이때 추가, 삭제에 의해 문서의 유효성이 유지되는 경우만을 가정하여 반복부에 대하여 추가, 삭제가 가능한 권한기술 모델을 제안하고 그 모델을 이용하여 웹 포탈사이트에 적용한 시스템의 시나리오를 제시하였다.

시스템 시나리오에서는 문서에 권한정보를 추가하는 Server와 권한이 추가된 문서를 분석하여 사용자에게 View를 제공하는 Client를 제시하였는데 통신상의 보안에 대하여서는 고려하지 않았다. 또한 본 논문에서는 문서의 유효성이 유지되는 경우만을 가정하여 연구하였으므로 향후 문서의 유효성을 변경하는 추가, 삭제를 고려한 모델과 XForms를 이용하여 실제 시스템을 구축하는 연구를 진행하고자 한다.

참고문헌

[1] 최동희, 박석 "접근제어 정책구현을 위한 역할기반 XML 암호화". 정보보호학회논문지 15권 제1호. 2005, 2.

[2] Chung-Hwan Lim, Seog Park, Sang H. Son "Access control of XML documents considering update operations". Proc. of the 2003 ACM workshop on XML security. 2003

[3] Irini Fundulaki, Maarten Marx "Specifiying Access Control Policies for XML Documents with XPath",In The ACM Symposium on Access Control Models and Technologies, pages 61--69. ACM Press, 2004.

[4] Jason Crampton "Applying Hierarchical and RoleBased Access Control to XML Documents". Proc. of the 2004 workshop on secure web service SWS' 04. 2004

[5] Naizhen Qi,Michiharu Kudo, Jussi Myllymaki, Hamid Pirahesh "A function-based access control model for XML databases". Proc. of the 14th ACM International conference on Information and knowledge managemnet. 2005

[6] R.S. Sandhu, E. J. Coyne, H.L. Feinstein, and C.E. Youman "Role-based Access Control Models" IEEE Computer 29(2): 38-47. 1996

[7] W3C "www.w3.org/XML"