안전한 XML 접근 제어의 정책 설계에 관한 연구

A Study on Policy Design of Secure XML Access Control

조선문*, 주형석**, 유원희***
배재대학교*, 유한대학교**, 인하대학교***

Sun-Moon Jo(sunmoon@pcu.ac.kr)*, Hyung-Seok Joo(hsjoo@yuhan.ac.kr)**,
Weon-Hee Yoo(whyoo@inha.ac.kr)***

요약

접근 제어 기법은 아주 작은 단위의 보호 수준을 지원할 만큼 충분히 유연해야 한다. 또한 접근 제어 정책은 문서 타입과 관련하여 명세될 가능성이 매우 높으므로 문서가 기존의 접근 제어 정책에 의해 다루어지지 않는 상황을 적절하게 관리해야 한다. 기존의 접근 제어는 HTML의 근본적인 한계 때문에 정보 구조와 의미론을 충분히 고려하지 못하였다. 또한 XML 문서에 대한 접근 제어는 읽기 실행만을 제공하며, 복잡한 권한의 평가 과정 때문에 시스템 성능이 저하되는 문제점이 존재한다. 이러한 문제점을 해결하기 위해 본 논문에서는 미세 접근 제어가 가능한 XML 접근 제어 관리 시스템을 설계한다. 접근 제어 시스템을 개발함에 있어 어떤 권한부여를 명세하고 어떤 접근 제어를 실행해야 하는가와 관련하여 XML 문서 권한부여 주체와 객체 정책에 관하여 기술한다.

■ 중심어 : | XML | 권한부여 | 정책 |

Abstract

Access control techniques should be flexible enough to support all protection granularity levels. Since access control policies are very likely to be specified in relation to document types, it is necessary to properly manage a situation in which documents fail to be dealt with by the existing access control policies. The existing access control has not taken information structures and semantics into full account due to the fundamental limitations of HTML. In addition, access control for XML documents allows only read operations, and there exists the problem of slowing down system performance due to the complex authorization evaluation process. In order to resolve this problem, this paper designs a XML Access Control Management System which is capable of making fined-grained access control. And then, in developing an access control system, it describes the subject and object policies of authorization for XML document on which authorization levels should be specified and which access control should be performed.

■ keyword : | XML | Authorization | Policy | Access Control |

* 본 논문은 2006학년도 배재대학교 교내학술연구비 지원에 의하여 수행된 것입니다.

접수번호 : #071008-003
접수일자 : 2007년 10월 08일
심사완료일 : 2007년 11월 06일
교신저자 : 조선문, e-mail : sunmoon@pcu.ac.kr

I. 서 론

XML(eXtensible Markup Language)은 인터넷상에서 논문, 의학, 경영 등 복잡하고 구조화된 문서자료의 저장, 관리, 검색을 용이하게 할 수 있을 뿐만 아니라 전자상거래, 전자 도서관, 가상대학 등의 핵심 응용 시스템의 구축에서 중요한 역할을 하고 있다. 인터넷 상에서의 데이터 교환 및 표현의 표준으로 사용된 이후, 많은 새로운 데이터들이 XML 타입으로 작성되고, 기존의 데이터들이 XML 타입으로 변환되어 현재는 XML 형태의 데이터양이 크게 증가하고 있다[1]. XML은 스스로 의미 있는 정보를 기술할 수 있는 장점을 이용해 기업체의 데이터베이스나 응용 프로그램의 운영중에 생기는 많은 데이터들에 대한 정보 교환 형태의 표준 데이터 타입을 제공할 수 있다. 따라서 상세 정보와 의미를 정의하고 기술할 필요가 있는 컴포넌트 명세나 문서 관리 시스템 등에 매우 적합하다.

XML 문서는 민감도가 다양한 정보를 포함하여 아주 작은 단위의 접근 보호 수준을 지원해야 한다. 어떤 경우에는 여러 문서에 같은 접근 제어 정책이 적용될 수 있고, 어떤 경우에는 같은 문서의 부분마다 다른 접근 제어 정책이 적용될 수 있다. XML 문서는 보통 웹 사이트 기반으로 사용되고 있으므로 기존의 웹서버에 XML 문서의 접근 제어 시스템을 확장할 수 있어야 한다. XML 문서가 항상 사전 정의된 문서 타입에 맞는 것은 아니다. 접근 제어 정책은 문서 타입과 관련하여 명세될 가능성이 매우 높으므로 문서가 기존의 접근 제어 정책에 의해 다루어지지 않는 상황을 적절하게 관리해야 한다. 웹에서는 문서 교환 및 수집 과정이 빈번할 수 있으므로 이러한 상황에 적절히 대처할 수 있는 접근 제어가 요구된다.

기존의 웹 기반의 접근 제어는 파일 단위나 파일의 부분에 권한을 기술하는 것이 가능하다. 그러나 이런 방법은 XML 문서의 정보 의미에 기반한 접근이나 요소와 같이 아주 작은 단위의 접근이 불가능하다. 이러한 접근 제어 모델[4][5]들은 DOM(Document Object Model) 트리를 이용하여 XML 문서와 DTD의 요소에 접근 권한을 설정한다. 설정된 접근 권한 정보에 의해 사용자의 XML 데이터 접근을 제어한다.

기존의 접근 제어 모델은 XML에 대한 모델과 마찬가지로 데이터를 의미론적으로 구조화할 수 있는 언어를 기반으로 하지 않는다는 사실에서 유도되는 큰 한계가 있다. 따라서 안전한 권한부여를 관리하는 일이 매우 어렵다. 특히 문서의 부분에 접근하고자 할 경우 페이지를 수동으로 여러 부분으로 나누어 서로 다른 권한을 부여해야 한다. 또한 XML 문서 접근 제어는 각 연산에 따른 접근 제어 기법이 복잡하다. 권한부여 설정과 DTD 검증 과정에서 XML 문서의 파싱 작업과 DOM 트리의 반복적인 검색 때문에 많은 메모리를 사용하여 시스템 성능이 저하되는 문제점도 발생한다[2-5][8].

XML 기반 접근 제어는 인터넷상의 접근 제어 서비스를 위해 서로 다른 환경에서 일관되게 적용될 수 있는 권한부여 정책을 제공하고 정책을 통하여 기존의 다양한 환경에 상호운영이 가능하도록 해야 한다. 본 연구는 XML 문서를 위한 미세한 접근 제어 시스템 개념을 제안한다.

II. 관련 연구

기존의 웹 기반의 접근 제어는 파일 단위에 권한을 기술하는데 가능한다. 그러나 XML 문서의 특징인 정보의 의미에 따라 처리하기 위한 정보 의미에 기반한 접근과 요소와 같이 아주 작은 단위의 보호 수준의 접근이 불가능하다. 따라서 XML 문서의 접근 제어를 위한 요구사항을 정리해 보면 다음과 같다[3][6][7][9].

- 기존의 웹 서버 기술의 확장이 가능해야 한다. XML 문서는 보통 웹 사이트 기반으로 사용되고 있으므로 기존의 웹서버에 XML 문서의 접근 제어 시스템을 확장할 수 있어야 한다.
- 접근 제어 시스템의 연산이 수행되는 것이 사용자에게 투명하여야 하며 요청자가 보는 문서 중 어느 문서에 권한이 주어지지 않아 거부되었는지 알 수 없어야 한다.
- 다중 사용자 환경에서는 동일한 스키마를 기반으

로 생성된 인스턴스 문서에 대하여 사용자별 접근 제어가 반드시 필요하다. 즉, 동일한 스키마를 따라 생성된 인스턴스 문서의 정보는 동일한 역할을 부여 받은 사용자 별로 다른 접근 제어를 수행할 수 있는 메커니즘이 필요하다.

- XML 문서는 항상 미리 정의된 문서 형태로 구성될 수 없다는 점으로 사용자의 요청에 따라 반환되어야 하는 문서 포맷은 사전 정의 없이 적절한 접근 제어의 적용이 가능한 메커니즘과 이를 지원하는 동적인 접근 제어가 필요하다.

Gabillon[4]는 권한부여 규칙을 4-튜플로 구성하였다. 권한부여 규칙은 주체의 집합, 객체의 집합, 접근, 우선권으로 구성된다. 이 모델에서 권한 부여 규칙이 가지는 의미는 노드의 타입과는 상관이 없이 표현했다. 만약 노드 n으로의 접근에 대한 승인이 사용자에게 주어진다면 사용자는 n이 서브 트리를 볼 수 있도록 허가한다. 만약 노드 n으로 접근에 대한 거부가 사용자에게 주어진다면 사용자는 n이 서브 트리를 볼 수 있는 것에 대하여 거부한다. [4]에서는 권한 부여의 의미론을 다양한 것으로 정의하였다. 그러나 이 모델은 모든 종류의 노드를 보호할 수 있는 가능성을 제공하지 않는다. 더구나 충돌 해결 정책은 복잡하다. 즉, 의미론은 보호를 받는 객체에 따라서 다양해진다는 것을 의미한다. 또한 XML 문서의 접근 제어는 읽기 연산만을 제공하고, 접근 제어 시스템의 구현에서 권한의 평가 과정이 복잡하고 응답시간이 느린 단점이 있다.

[9]는 갱신 연산을 지원하는 XML 모델을 제안하기 위해 XML 갱신 연산자를 정의했고, 이를 접근 제어 모델에 포함시켰다. 접근 제어에 갱신 연산자를 추가하면서 발생하는 성능상의 문제점을 해결하기 위해 새로운 액션 타입을 정의했다. 접근 제어 모델에서는 접근 제어의 과정을 2단계로 나누었다. 이와 같이 접근 제어의 과정을 2단계로 나눔으로써, 1단계에서 거부되는 연산자들로 인한 불필요한 작업이 제거할 수 있다고 하였다.

그러나 이 논문에서는 환경을 가정하여 연구하였다. 예를 들면, XML의 엘리먼트 사이에 의미적 종속성은 없다. 또한 검색보다는 갱신 질의가 매우 빈번하게 발생한다. 그리고 이 논문은 검색 질의에 대해서 오버헤드가 만이 발생하는 문제점도 있다.

XrML[12]은 디지털 자원 활용에 대한 권리와 조건을 표현하기 위한 XML 기반이다. XrML은 직접적인 방식으로 다양한 작업 흐름 단계에서 단순한 권리와 복잡한 권리를 모두 표현할 수 있다는 점에서 포괄적이다. 그러나 XrML은 본 논문의 XML 문서 접근을 위한 보안 요구사항을 고려하고자 할 때, 다음과 같은 문제가 있다.

첫째, XrML은 주체의 그룹 구성원 같은 매우 단순한 특성을 투명하게 표현하는 데 어려움이 있으며 목표 문서의 요소가 요구 제기자의 특정 속성과 부합되어야 한다는 제약이 있다.

둘째, XrML은 전자 서적, 오디오, 비디오 파일 같은 정적 자원을 다루는데 더 적합한 반면, 수정 가능한 XML 문서와 같은 동적 자원을 다루는 데에는 어려움이 있다.

III. XML 문서들을 위한 접근 제어 정책 설계

1. XML 문서 권한부여 주체 정책

일반적으로 주체는 식별 번호나 요청이 나온 위치를 토대로 언급할 수 있다. 위치는 숫자로 된 IP 주소(예: 203.246.43.87)나 심볼릭 이름(예: lab.pcu.ac.kr)과 관련하여 나타낼 수 있다. 본 논문에서는 IP 주소, 심볼릭 이름을 사용한다. 따라서 접근을 요청하는 주체는 사용자 ID, IP 주소, 심볼릭 주소로 구성된다. 여기서 사용자 ID는 사용자가 연결한 서버의 사용자 ID를 의미하며, IP-주소와 심볼릭-주소는 사용자가 서버에 접속한 머신을 의미한다.

사용자들과 머신에 적용할 수 있는 권한부여 명세를 허가하기 위해 본 논문에서는 사용자 그룹과 위치 패턴을 지원한다. 사용자 그룹은 서버에 정의된 사용자들의 집합이다. 위치 패턴은 기호나 숫자로 된 식별자와 관련하여 물리적 위치를 식별하여 표현한다. 패턴은 특정 명칭이나 숫자 대신 와일드카드(*) 문자를 이용하여 명세한다.

사용자와 그룹은 멤버십 관계와 함께, IP 주소는 패턴과 함께, 심볼릭 이름은 패턴과 함께, 부분적으로 순서화된 집합을 가지고 있다. 주체의 다양한 구성요소를 일정하게 처리하기 위해 본 논문에서는 정의 1과 같이 계층을 제안한다.

정의 1. 계층

- 집합을 대문자 A, B, C,…, X, Y, Z 등으로 표시한다.
- 요소를 소문자 a, b, c,…, x, y, z 등으로 표시한다.
- X, Y를 임의의 집합이라 하면, X×Y의 임의의 부분집합R을 X로부터 Y의 관계라고 정의한다.
- X×X의 임의의 부분집합 R, 즉 R⊂X×X를 X에서의 관계라 정의한다.
- 집합 X에서의 관계 R이 반사 관계, 반대칭 관계, 추이 관계가 성립하면 관계 R을 부분 순서 관계(Partial order relation)라 정의한다(R이 X에 대한 부분 순서 관계이면 순서쌍 (X, R)을 부분순서 집합(Partially ordered set)이라고 표시한다).
- R을 집한 X에서의 관계, 즉 R⊂X×X라 하면, X의 모든요소 x에 대하여 (x, x)∈R이면 R은 반사적이라 정의한다.
- R을 집합 X에서의 관계, 즉 R⊂X×X라 하면, (x, y)∈R이고(y, x)∈R 일 때 x=y이면 R은 반대칭적이라 정의한다.
- R을 집합 X에서의 관계, 즉 R⊂X×X라 하면, (x, y)∈R이고(y, z)∈R 일 때 (x, z) ∈R이면 R은 추이적이라 정의한다.

본 논문의 접근 제어 관리 시스템은 다음을 고려한다.

- 사용자 그룹은 UG=(U, UG, ≤UG)이다.
  U는 사용자 식별자의 집합이고, G는 사용자 그룹 이름의 집합이면, UG = U ∈ G이다. 두 요소 x, y ∈ UG, x ≤UG y로 주어진다.
- IP(internet protocol)는 IP = (I, IP, ≤IP)이다.
  I는 완전하게 명세된 숫자로 이루어진 주소의 집합이다. IP는 IP 패턴의 집합이다. y의 각 요소가 와일드카드 문자이거나 x의 해당 요소와 같을 경우, 두 요소 x, y ∈ IP, x≤IP y로 주어진다.
- 심볼릭 이름(symbolic name)은 SN = (S, SN, ≤SN)이다.
  S는 완전한 심볼릭 이름이며 SN은 심볼릭 이름 패턴의 집합이다. y의 각 요소가 와일드카드 문자이거나 x의 해당 요소와 같을 경우, 두 요소 x, y ∈ SN, x ≤SN y로 주어진다.

2. XML 문서 보안 권한부여 객체 정책

XML 문서를 미세한 보호 수준의 요구사항을 시행하기 위해 권한부여 명세는 XML 문서 집합에서 문서의 특정 부분에 이르기까지 광범위한 보호 객체들을 지원해야 한다. XML 문서의 경우 URI는 path expressions으로 확장할 수 있는데 이것은 문서 내에서 요소와 속성을 식별하는데 사용된다. URI는 XML 문서를 보호해야 할 자원을 표시한다.

본 논문에서는 XML 문서의 내부 구성요소 식별을 위해 W3C에서 제안한 XPath 언어를 사용한다[3]. 표준 언어를 도입하므로 다음의 장점이 있다. 첫째, 언어의 구문과 의미론을 사용자가 잘 알고 있다. 둘째, 함수 시스템을 만들기 위해 쉽게 재사용할 수 있다. 또한 XPath는 문자열, 숫자, 부울 논리, 노드작업을 조작할 수 있는 많은 함수를 제공한다.

Ⅳ. 접근 권한부여 메커니즘

XML은 계층적인 트리 구조이기 때문에 상위 노드의 권한이 하위 노드의 권한에 영향을 미칠 수 있다. 동일한 사용자라 하더라도 속해있는 그룹, IP 주소, 컴퓨터 이름에 따라서 권한이 달라질 수 있다. 주체가 부호는 다르지만 같은 보호 객체에 대한 같은 권한을 놓고 두 가지 권한이 부여된다는 점에서 권한 부여 사이에 충돌이 발생한다. 본 연구의 시스템에 의해 실행되는 충돌 해소 정책은 정의 1과 같은 원리를 토대로 권한의 우선순위를 결정하는 규칙들을 제안한다.

정의 2. 권한 충돌 우선순위 규칙

step1: 주체 관계사이의 부분적 순서에 의해 기술된 가장 상세히 기술된 주체에 관한 권한에 우선

순위가 높다.

step2: 전파되어 발생한 권한보다는 직접 기술된 권한이 우선순위가 높다.

step3: DTD에 기술된 권한보다는 XML 문서에 직접 기술된 권한이 우선순위를 갖게 된다.

step4: 노드상의 권한이 그의 조상의 권한 보다 우선시 된다.

레이블링은 보안 관리자가 정의한 접근 권한 정보를 이용하여 사용자 질의에서 요청하는 DOM 트리의 노드에 접근 권한을 설정하는 과정이다. 레이블링된 권한정보는 사용자의 질의를 거부 또는 허가할지 결정하는데 사용된다. 연산자 단위로 권한 정보를 DOM 트리에 레이블링 한다면, 질의문에 포함된 연산자의 종류 수만큼 레이블링을 반복하여 처리한다. 이러한 반복적인 레이블링을 제거하기 위해 본 논문에서는 접근 제어 관리 알고리즘을 제안한다.

```
rq : Requester(subject, object, action, altg, sign, type),
xml : XML Document URI,
dtd : DTD of XML,
ap : Authorization Policy(auth.dtd, xml.xas)
T : Dom Tree
begin
        T ← Build Dom Tree from xml
        T ← Initial_Label(T, rq, ap)
        T ← Label(T, rq, ap)
        T ← Prune(T)
end
```

그림 1. XML 접근 제어 관리 알고리즘

[그림 1]은 XML 문서 보안을 위한 XML 접근 제어 관리 알고리즘이다. 이 알고리즘은 4단계로 구성된다. 첫 단계는 XML 문서에 관한 DOM tree를 구성하고 두 번째 단계는 DOM tree에 대한 접근 제어 초기화 레이블링을 수행한다. 세 번째 단계는 각 노드에 대해 권한 설정과 권한의 충돌을 해결하는 단계이다. 마지막으로 최종 권한 설정 정보를 가지고 문서를 제거하는 과정으로 구성된다.

[그림 1]에서 사용 의뢰자, XML 문서(XML Document URI), XML의 DTD(DTD of XML), 안전한 접근 권한 부여 정책 등을 입력 값으로 사용한다. 그리고 ap는 auth.dtd와 xml.xas로 구성된다. 전체 접근 권한부여 정보에서 XML 문서에 해당하는 접근 권한을 xml.xas로 하고, DTD에 해당하는 접근 권한을 auth.dtd로 사용한다.

```
Input:   T : Dom Tree,rq : requester,
           ap: Authorization Policy(auth.dtd, xml.xas)
Output:  T : Dom Tree
Procedure Initial_Label
 begin
    if  T.root then
        if  ap.auth.dtd ∪ ap.xml.xas ==∅   then
            T.root.label ← default()
        else
            T.root.label ← decision_rule(ap.auth.dtd, ap.xml.xas)
        fi
    fi
 end
```

그림 2. 접근 제어 초기화 레이블링 알고리즘

[그림 2]는 의뢰자와 XML 문서 Dom Tree가 있을 때, 그리고 ap가 먼저문서를 나타내는 트리로 변수 T를 초기화하고 T에 root를 초기화한다. 초기화의 목적은 권한부여를 설정하는 요소 또는 속성과 관련된다. 문서에 권한부여가 모든 의뢰자에게 적용되는 것은 아니다. 문서의 요소에 대한 권한부여와 트리에 따른 권한부여 설정은 의뢰자마다 다를 수 있다. 따라서 Initial_Label의 단계는 의뢰자에게 적용되고 인스턴스 및 스키마 수준에서 문서 URI에 대해 권한을 설정한다.

```
Input:  T : Dom Tree, rq : User_requester, ap : AP
Output: T : modified DOM Tree
Procedure Label
 begin
    for each c ∈ children(T.root) do
        if c.parent.type in (L, R, LD, RD) then
            if auth.dtd ∩ xml.xas ==  ∅  then
                c.label ← c.parent.label
            else
            c.label ← decision_rule(c.p.label, ap.auth.dtd, ap.xml.xas)
            fi

        else if ap.auth.dtd ∩ ap.xml.xas ==∅  then
                c.label ← default()
            else
            c.label ← decision_rule(ap.auth.dtd, ap.xml.xas)
            fi
```

```
        fi
    od
end
```

그림 3. 권한 설정과 충돌 해결 알고리즘

현재 노드가 트리의 루트인 경우 ap에 존재하는 auth.dtd와 xml.xas의 합집합 값이 ∅인 경우라면, 즉 접근 권한이 명시적으로 설정이 없다면, 미리 정해진 기본 접근 권한 값을 설정한다. 그렇지 않다면 명시적 권한 중 우선순위가 가장 높은 권한을 설정한다. default()는 보안 관리자가 특정 XML 문서에 명시적으로 권한 설정이 없을 경우에 기본 접근 문서만 제공할 수 있도록 권한을 설정한다. auth.dtd와 xml.xas의 합집합 값이 ∅이 아닌 경우라면, decision_rule()는 동일한 모드에서 충돌이 발생한 경우 미리 정해진 충돌 해결 규칙에 의해 우선순위가 가장 높은 권한을 설정한다.

[그림 3]에서 루트 노드의 경우는 부모 노드가 없지만 루트 노드를 제외한 노드는 부모 노드가 존재한다. 노드와 관련된 레이블은 하위 요소과 속성으로 전파된다. 타입은 로컬 권한부여, 재귀 권한부여, 로컬 권한부여는 스키마 수준에서 상세, 재귀 권한부여는 스키마 수준에서 상세 등으로 구성된다. ∅은 권한부여가 설정이 없을 경우이다. 각 자식에 대해 c 부모의 타입이 L, R, LD, RD에 해당하는 경우라면 AP로부터 ap.auth.dtd ∩ ap.xml.xas == ∅인지 확인하고 ∅인 경우라면 c의 부모의 레이블 정보를 c의 레이블에 배정한다.

```
Input:  T : Dom Tree
Output: T : Pruned DOM Tree
Procedure Prune
 begin
      post ← Postorder(T)
   for each n∈ post do
      if   n.children ==∅  & n.label  '+'  then
           remove n from T
      fi
  od
end
```

그림 4. 문서 제거 알고리즘

[그림 4]는 문서에 거부와 또는 권한이 설정 되지 않은 레이블이 붙은 노드를 포함하는 모든 서브 트리를 제거한다. 트리를 후위 탐색으로 방문하면서 현재 노드가 '+'가 아닌 경우라면 제거를 한다.

V. XML 접근 제어 관리 시스템의 설계

본 논문에서는 현재 아파치의 Xalan 툴로 발전한 DOM API의 자바 구현 서비스를 이용하여 자바로 프로토타입을 설계하였다. 본 논문에서 제안한 XACMS(XML Access Control Management System)를 구현하기 위한 도구는 CPU Pentium IV 2.4GHz, 하드디스크 80GB, 메모리 256MB, Windows XP 운영체제에서 아파치 XML 파서, DOM, 인터넷 익스플로러 6.0, Java 5.0이다.

XACMS의 구조는 [그림 5]와 같다. 사용자는 원격 사이트에서 XML 문서를 요청하면 원격 사이트의 XACMS는 사용자의 권한과 요청에 따라 XML 문서를 돌려준다. 보안 프로세서는 사용자에 의해 요청한 유효한 XML 문서와 인스턴스 수준에서 권한이 기술된 접근 제어 목록을 입력으로 한다. 또한 프로세서의 연산은 문서의 DTD와 스키마 수준에서 기술된 접근 제어 목록도 포함한다. 프로세서의 출력은 사용자에게 접근이 허가된 정보만을 포함하는 유효한 XML 문서이다. 시스템에서 문서와 DTD는 DOM 기술에 따라 내부적으로 표현된다.

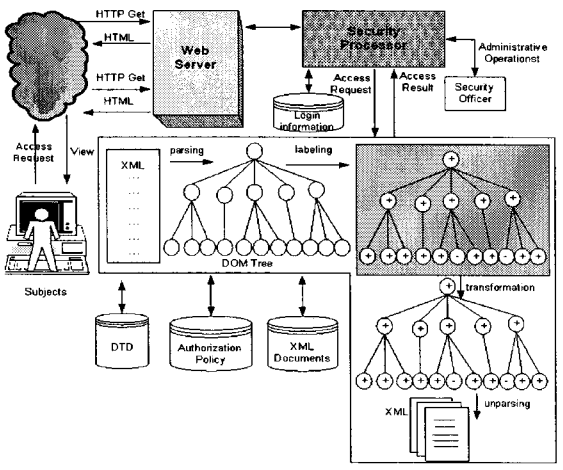


그림 5. XML 문서 보안을 위한 접근 제어 관리 시스템

1. 접근 제어 성능 평가

본 논문에서 제안한 XACMS는 검색 연산뿐만 아니라 XML 문서의 변경과 구조의 변경 모두 허가하는 환경을 지원하므로, 유효한 XML 문서와 잘 정의된 XML 문서까지 접근 제어의 대상이 될 수 있다.

성능 평가의 대상은 [3]에서 제안한 XML 접근 제어 기법과 XACMS 기법에 대하여 접근 가능성 비율을 비교하였다. XML 문서와 DTD는 [13]에서 XML 벤치마크[TXML bp]로부터 XML 데이터와 문서를 이용하였다.

첫 번째 실험은 접근 제어 데이터를 위하여 시드로 문서에 몇 개의 노드를 임의로 선택한 다음 시드에 접근 가능이나 접근 불가능이라는 레이블을 붙임으로써 사용자 접근 모드 합성으로 XML 데이터에 대한 접근 제어를 생성했다.

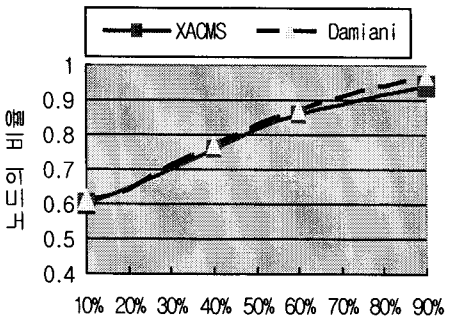


그림 6. 접근 가능성 비율

Damiani[3]는 접근 제어에 대한 레이블링 상태이므로 XACMS와 Damiani를 비교하였다. 다양한 접근 가능성비로 합성 접근 통제가 사용되는 약 15,000개의 노드에 대한 [TXMLbp] 문서를 이용하였다. 본 연구의 성능 측정 기준은 Damiani 노드 수와 XACMS 노드 수의 비이다.

[그림 6]은 접근 가능성 비가 10%에서 90%까지 달라짐에 따른 비교를 보여준다. 이러한 다양한 접근 가능성으로 접근 비를 비교하였다.

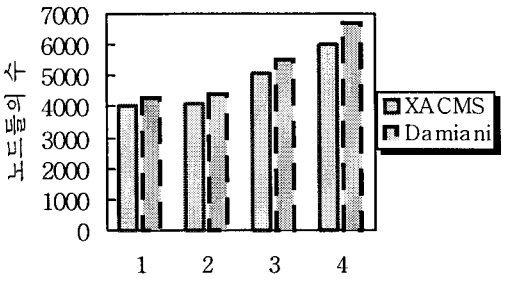


그림 7. 주체에 관한 변화 노드

두 번째 실험에서는 [그림 7]에서 네 개의 읽기 실행 모드 각각에 대해 많은 사용자를 표본 추출하고 각각의 단일 사용자에 대해 Damiani와 XACMS를 구성했다. 평균 사용자에 대한 XACMS 레이블 수와 Damiani 노드 수의 비는 [그림 7]과 같다. Damiani 노드와 XACMS 노드의 결과이다. 이것은 Damiani 노드와 XACMS 변환 노드가 같은 크기라고 가정하기 때문이다. 그러나 실제로는 XACMS 노드가 훨씬 작다. 이것은 Damiani가 접근 권한을 데이터에 따로 저장하기 때문이다. 그러므로 각각의 Damiani 모드에는 접근 제어 정보뿐 아니라 문서 노드 참조와 Damiani에 있는 노드의 자식에 대한 정보가 포함될 것이다. 이와 반대로, 문서 부호화에 접근 제어 정보를 편승시키는 XACMS는 변환 노드 당 접근 제어 부호를 하나만 저장한다. 따라서 Damiani가 XACMS보다 노드가 적더라도 Damiani에 요구되는 총 공간은 매우 크다.

VI. 결론

본 논문은 XML 문서 접근을 위한 권한부여와 효율적인 문서 관리를 위한 접근 제어 메커니즘을 정의하고 설계하였다. 기존 XML 문서의 보안에 관한 접근 제어는 데이터를 의미론적으로 구조화할 수 있는 언어를 기반으로 하지 않는다는 문제점이 있어 안전하게 권한부여를 관리하는 일이 매우 어려웠다. 또한 접근 제어는 각 연산에 따른 접근 제어 기법으로 권한부여 설정 및 DTD 검증 과정에서 XML 문서의 파싱 작업과 DOM 트리의 반복적인 검색 때문에 많은 메모리를 사용하여

시스템 성능이 저하되는 문제점도 발생하였다.

본 논문에서는 보안을 위한 XML 접근 권한부여 정책을 제안하고 효율적인 문서 관리를 위한 권한부여 전파 규칙과 XML 접근 제어 관리 알고리즘을 기술하였다. 또한 XML 문서를 위한 미세한 접근 제어 XACMS를 제안하고, 브라우징과 사용자를 위한 접근 모드를 제공하였다. 이를 통해 보안 관리자는 사용자가 요소에 있는 정보를 읽거나, 요소에 링크를 추가, 수정, 삭제할 수 있는 권한을 설정한다. 사용자들과 머신에 적용할 수 있는 권한부여 명세를 허가하기 위해 사용자 그룹과 위치 패턴을 지원하도록 설계하였다.

향후 연구로는 XML 문서를 이용하는 다른 응용에 각각의 특성을 반영하여 통합된 프레임웍을 제시하는 XML 접근 제어에 관한 연구가 필요하다.

참 고 문 헌

[1] T. Bray, *Extensible Markup Language(XML) 1.0*, World Wide Web Consortium (W3C), 2000.

[2] S. Hada and M. Kudo, "XML Access Control Language: Provisional Authorization for XML Documents," pp.1-28, 2002.

[3] E. Damiani, S. Vimercati, S. Paraboschi, and P. Samarati, "Design and implementation of an access control processor for xml documents," In proceedings of the 9th International WWW Conference, Amsterdam, 2000.

[4] A. Gabillon and E. Bruno, "Regulating access to XML documents," In Proceedings of the Fifteenth Annual IFIP WG 11.3 Working Conference on Database Security, 2001.

[5] S. Hada and M. Kudo, "XML Access Control Language: Provisional Authorization for XML Documents," 2000.

[6] E. Bertino, M. Braun, S. Castano, E. Ferrari, and M. Mesiti, "Author-X: A Java-Based System for XML Data Protection," Technical report, Dipartimento di Scienze dell'Informazione, University of Milano, submitted for publication, 2000.

[7] S. M. Jo, K. T. Kim, H. J. Kouh, and W. H. Yoo, "Access Authorization Policy for XML Document Security," Proceedings of International Symposium on Parallel and Distributed Processing and Applications ISPA Workshops 2005, Vol.3759, pp.589-598, 2005.

[8] E. Bertino and E. Ferrari, "Secure and Selective Dissemination of XML Documents," ACM Transactions on Information and System Security, Vol.5, No.3, pp.290-331, 2002.

[9] C. H. Lim, S. Park, and S. H. Son, *Access Control of XML Documents Considering Update Operations*, In Proceedings of the 10th ACM workshop on XML security, Fairfax, VA, USA, 2003.

[10] X. Zhang, J. Park, and R. Sandhu, "Schema based XML Security: RBAC Approach," IFIP WG 11.3 Working Conference on Data and Applications Security, pp.300-343, 2003.

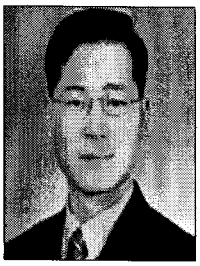
[11] http://www.w3.org/TR/xpath20

[12] http://www.xrml.org

[13] A. R. Schmidt, F. Waas, M. L. Kersten, D. Florescu, I. Manolescu, M. J. Carey, and R. Busse, *The XML Benchmark Project*, Technical Report INS-R0103, CWI, Amsterdam, The Netherlands, 2001.

저 자 소 개

조 선 문(Sun-Moon Jo) 정회원



- 1999년 2월 : 충주대학교 컴퓨터공학과 (공학사)
- 2001년 2월 : 인하대학교 전자계산공학과 (공학석사)
- 2007년 8월 : 인하대학교 컴퓨터정보공학과 (공학박사)
- 2006년 3월 ~ 현재 : 배재대학교 IT 교수

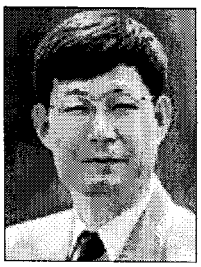
<관심분야> : XML, 정보보안, 임베디드 시스템, 프로그래밍 언어

주 형 석(Hyung-Seok Joo) 정회원

- 1985년 2월 : 인하대학교 전자계산학과 (이학사)
- 1987년 2월 : 인하대학교 전자계산학과 (공학석사)
- 1997년 2월 : 인하대학교 전자계산공학과 (공학박사)
- 1987년 9월 ~ 현재 : 유한대학 컴퓨터정보과 교수

<관심분야> : 함수언어, XML, 정보보안

유 원 희(Weon-Hee Yoo) 정회원



- 1975년 2월 : 서울대학교 응용수학과 (이학사)
- 1978년 2월 : 서울대학교 대학원 계산학 (이학석사)
- 1985년 2월 : 서울대학교 대학원 계산학 (이학박사)
- 1979년 ~ 현재 : 인하대학교 컴퓨터정보공학 교수

<관심분야> : 컴파일러, 프로그래밍 언어, 병렬시스템