웹서비스 통합 인증에서의 XACML적용 모델 연구

박병철*, 유석환, 신동규, 신동일, 박범대**, 김형순**
세종대학교 컴퓨터공학과, 한국전산원**

A Study on the XACML Model for Integrated Authentication in Web Services

ByungChul Park*, SeokHwan Yu, DongKyoo Shin, DongIl Shin, BeomDae Park**, HyongSoon Kim**
Department of Computer Engineering, Sejong University, National Computerization Agency**

요 약

웹서비스에 대한 높은 관심과 함께 그 실현이 점차 가시화 되고 XML로 이루어진 지원 기술들의 전폭적인 지지로 잠재력은 더욱 확대되고 있다. 웹서비스는 기존 웹 기반의 디스플레이에 그쳤던 단순정보 교환을 애플리케이션 차원에서 데이터를 통신할 수 있어 개발 가능성이 무한한 프레임워크로 각광받고 있다. 그러나 현재 자원 관리에서의 효율성 문제가 드러나게 되었고 인증분야와의 접목에서도 한계를 드러내고 있다. XACML은 리소스에 대한 미세한 접근제어를 할 수 있는 XML기반의 언어이다. 접근 하려는 리소스는 XML을 사용하여 표현되는 어떠한 객체도 될 수 있으며XPath나 LDAP과 같은 프로토콜과 함께 바인딩하여 사용될 수도 있다. 보다 효율적인 접근 제어를 위하여 XACML적용 모델을 연구하였다.

I.서론

인터넷을 통한 e-business의 시장이 그 규모와 영역에 있어서 매년 기하급수적으로 성장하고 있다. 이러한 추세 속에서 최근 가장 주목 받고 있는 신기술은 바로 '웹서비스(Web Services)'이다.

웹서비스는 기존 인터넷 프로토콜을 사용할 수 있어 기본 재반비용이 작아지고 XML 기반의 SOAP 인터페이스를 사용해 접근할 수 있는 어플리케이션으로 확장성과 유연성이 뛰어나다. 하지만, 효율적 자원 관리와 자원 사용의 권한 설정이 명확히 정의되지 않고 있어 범용적으로 e-business에 적용되지 못하고 있다. 이에 본 논문에서는 W3C와 OASIS의 주도하에 XACML 기반 정보 보호 기술을 적용하여 웹서비스 보안 강화에 대한 방안 연구를 하였다.

II.관련 연구

1.Web Services

1)Web Services의 개요

지금까지 제공되어온 일반적인 웹서비스와는 달리 Web Services는 표준 RPC를 통해 프로토콜에 의존적이지 않도록 배치되어 바인딩 될 수 있는 비즈니스 분산 객체이다. 표준 인터넷 프로토콜인 SOAP을 사용하여 기존의 HTTP와 같은 인터넷 프로토콜을 그대로 사용하므로 재반 비용이 적어지고 XML을 기반으로 하기 때문에 확장성과 유연성이 있다. 캡슐화된 어플리케이션으로 웹에 존재하는 컴포넌트의 재조립으로 새로운 웹서비스로의 구성이 가능하다. 웹서비스의 기본 구성 요소는 서비스 요청자, 제공자, 레지스트리로 나눌 수 있는데 서비스 제공자는 레지스트리에 제공하는 웹 애플리케이션 객체를 등록하고 요청자는 레지스트리에서 서비스를 검색하여 원하는 서비스를 찾을 수 있고 WSDL을 사용하여 서비스와 통신하기 위한 모듈을 생성하여 제공자와 직접 통신할

수 있게 된다[1].

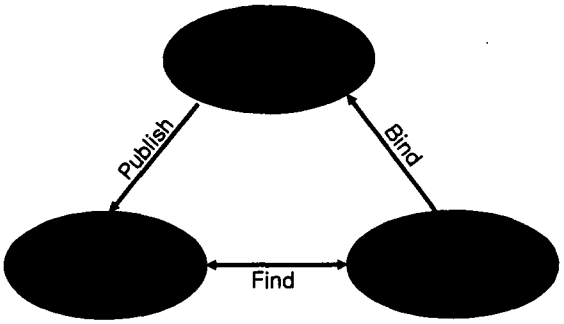

그림 1: 웹서비스 아키텍처

2)Web Services의 핵심 기술

■SOAP(Simple Object Access Protocol)

XML기반 프로토콜로 복잡한 객체 데이터 타입도 쉽게 모델링 할 수 있게 해주며 RPC프로토콜을 지원한다. HTTP뿐 아니라 FTP, SMTP, POP3등 기존의 프로토콜 상에서 동작하므로 부가적인 비용이 발생하지 않으며 특정 벤더에 종속되지 않은 공개프로토콜로 웹서비스에서 사용되는 모든 메시지는 SOAP을 사용하여 통신한다 [2].

■WSDL(Web Service Description Language)

XML기반의 웹서비스 기술 스크립트 언어로 웹서비스에 접촉하고 이용하기 위한 메시지 스키마를 정의하고 있다. 웹서비스 제공자의 endpoint가 어떤 메서드, 속성, 인수, 리턴 값을 가지는지 알려주어 클라이언트에서의 모듈 생성을 가능하게 한다. 이는 자동으로 이루어질 수 있으며 서비스 구현에 따라 생성 방법은 다양할 수 있다 [3].

■UDDI(Universal Description, Discovery and Integration)

일종의 디렉토리 서비스로서 웹서비스의 제공자는 자신의 서비스의 기능을 기술하고 UDDI에 WSDL을 등록하게 된다. 서비스 요청자는 UDDI를 통해 등록된 웹서비스를 간단히 검색할 수 있으며 WSDL에 의한 클라이언트 생성으로 서비스 제공자와 통신할 수 있다 [4].

2.SAML (Security Assertion Markup Language)

SAML[8]은 OASIS의 STTC(Security Services Technical Committee)가 제안한 XML 기반의 인증(authentication) 및 승인(authorization) 정보를 안전하게 교환하기 위한 프레임워크이다.

그림 1은 SAML을 이용하여 시스템 엔터티가 접근 제한된 자원에 접근하는 유즈케이스(use-case)의 흐름을 나타낸 것이다. 우선, 보증정보(credential information)를 모아 credential assertion을 구성한다. 다음으로는 수집된 보증정보를 이용해 사용자를 인증하게 된다. 인증 시 authentication assertion을 전달하기 위해 외부 PKI 서비스를 이용할 수도 있다. 추가적인 요구에 따라 session assertion 또는 authorization decision assertion 단계로 진행된다.

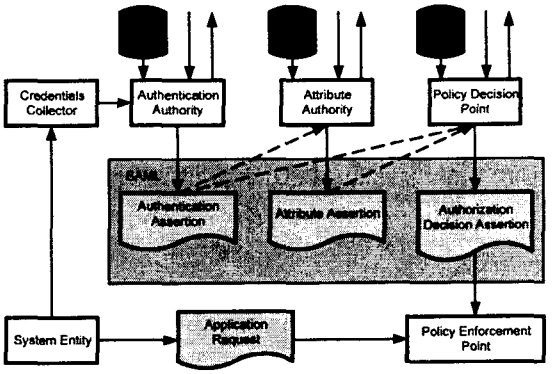

그림 2: SAML 아키텍처

3.XACML (XML Access Control Markup Language)

XACML[9]은 2003년 7월 24일 W3C에서 제안한 접근제어 리스트(access control list)를 통해 보안이 요구되는 자원에 대해 미세한 접근 제어 서비스를 제공할 수 있는 XML 기반의 언어이다. XACML은 SAML PDP(Policy Decision Point)의 일부로서 역할을 수행 할 수 있다. XACML의 정의에 따라 각각의 사용자 별 XML 문서 접근 정책을 수립하고 적용 할 수 있다.

XACML은 크게 object, subject, action의 3가지 요소로 구성되는데 subject는 사용자의 ID나 그룹, 또는 역할 등을 나타낼 수 있으며, object 요소는 subject가 접근할 데이터를 의미하며 그 데이터 참조로서 단일 XML 문서에서 개별 요소 수준까지 지정 할 수 있다. action은 4가지 수행 가능 동작으로 구성되며 각각은 읽기, 쓰기, 생성, 삭제 작업이다.

다이어그램에서 보여주는 데이터 흐름은 레포지토리에서 유용하게 사용될 수 있다. 예를 들어 PDP와 PIP, PDP와 PRP 혹은 PAP와 PRP사이의 통신은 레포지토리에 유용할 수 있다. 그러나 XACML 명세는 레포지토리에서의 사용이나 특정한 통신 프로토콜에서의 활용에 제한을 두고 설계되지는 않았다. 모델에서 보여주는 데이터의 흐름은 다음과 같은 순서를 따른다.

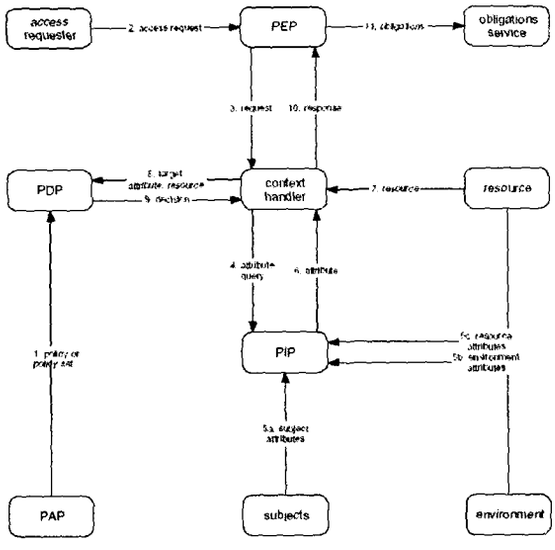

그림 3: 데이터 흐름 모델

1.PAP는 policy statements를 만들어서 PDP에서 사용가능 하도록 한다.

2.접근자는 PEP에 접근 하기위한 요청을 한다

3.PEP는 context handler에 접근 요청을 전송한다.

4.PIP는 attributes를 요청 할 수 있다.

5.PIP는 요청된 attributes를 얻는다.

6.PIP는 요청된 attributes를 context handler로 반환한다.

7.선택적으로 context handler는 context에 리소스를 포함한다.

8.context handler는 PDP에 decision reguest를 전송한다.

9.PDP는 context handler에 권한 결정을 포함한 응답 context를 반환한다.

10.context handler는 PEP에 응답을 반환한다.

11.context handler는 응답 context를 PEP형식에 맡게 변환하고 PEP는 프로세스를 이행한다.

12.접근이 허용되면 PEP는 리소스의 접근을 허용한다. 허용되지 않으면 접근이 거부된다.

III.웹서비스의 보안 구조

그림 2는 웹서비스 보안 아키텍처를 보여주고 있다. 여기서, 서비스 요청자와 제공자 사이의 통신은 SOAP을 이용하며, XML 전자서명과 암호화를 통하여 통신 메시지의 기밀성, 무결성, 부인방지가 이루어진다. 이 때, 서명과 암호화에 사용되는 키의 관리는 XKMS를 통해 수행된다.

자원 관리와 권한 설정을 위해 SAML, XACML이 적용된다. SAML을 통해 요청자는 ID, 속성, 권한 정보 등 인증 전반의 정보를 서비스 제공자에게 보내고 XACML이 이를 가지고 각 서비스 정책에 따라 권한을 결정, 요청자에게 부여한다.

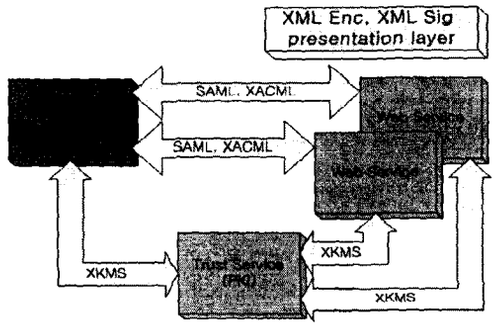

그림 4: 웹서비스 보안 아키텍처

IV.자원 접근 관리 표준 및 기술 동향 연구

자원 접근 관리는 자원에 허가받지 않은 접근을 차단하여 시스템의 안전성을 보장해 주는 것이다. 자원접근은 특성에 따라 3가지의 관리 방법이 있다.

■User ID based Access Control

사용자 ID에 기반 자원에 대한 접근 관리 방법으로 가장 많이 사용되고 있는 방법이다. 하지만 ID만을 사용한 자원접근 관리는 사용자의 증가에 따른 ID 관리와 확장성에 문제가 있다.

■Role-based Access Control

역할에 기반을 둔 자원에 대한 접근 제어 방법이다. 각 사용자에게 사용자의 자격에 맞는 역할들이 할당되고, 각 역할에 대하여 권한이 설정된다.

■Attribute-based access Control

속성에 기반 자원에 대한 접근 제어 방법으로 Group, Role, Clearance 등과 같은 다양한 속성을 설정하여 권한 관리한다.

(가)자원 접근 관리의 핵심 용어

■정책 (Policy)

정책은 하나 이상의 접근 제어 규칙과 규칙을 적용할 사용자나 그룹, 그리고 성공적인 승인 후에 애플리케이션에 제공될 응답의 집합으로 구성된다. 한 정책은 여러 그룹에게 동일한 규칙을 적용할 수 있다. 서로 다른 규칙을 정의해서 만들어진 서로 다른 수준의 승인을 이용한다면, 웹 서버의 동일한 영역을 보호하기 위해 여러 가지 정책을 사용할 수 있다. 정책은 단지 IP주소의 집합과 연관을 맺을 수 있어서 지역 기반의 접근 제어 기능을 제공할 수 있다. 그리고 한 그룹에서 특정 개인을 제외할 수 있는 정책을 지원할 수 있으면 좀 더 효율적인 사용자 관리를 할 수 있다. 이 기능은 수많은 사람을 포함한 그룹에 대한 정책을 쉽게 생성하도록 한다. 이러한 기능이 없다면 관리자가 사용자 그룹의 대규모 하부 조직에 대한 정책을 세우기가 어려울 것이다.

■자원 (Resource)

사용자가 접근을 시도하는 논리적, 물리적 오브젝트, 전형적으로 웹 페이지, 애플리케이션, CGI 스크립트 또는 디렉터리가 될 수 있다. 어떤 HTML 문서나 스크립트도 개별적으로 인가될 수 있다.

■규칙 (Rule)

규칙(rule)은 특정 자원에 대한 접근 허가를 규정하는 방법을 제공한다. 규칙은 한 자원에 대해 어떤 행위가 수행될 수 있고, 언제 그 행위가 수행될 수 있는 허가된 시간인지를 결정한다.

(나)자원과 역할(role)에 대한 관리

관리자가 요구하는 보안 모델 요구 사항에 대한 부분에서 기술된 보안 모델 테이블과 유사한 테이블이나 차트를 사용해서, 자원(resources) 컬럼에 자원을 리스트한다.이 경우, 단일한 자원에 대해 모든 부속 자원을 식별한다. 예를 들어, /secretary라는 디렉터리를 서브 디렉터리로 가지고 있는 /bose디렉터리에서 /secretary디렉터리는 자원으로써 모두 리스트 되어야 한다. 각각의 서브 디렉터리를 분리된 자원으로써 관리한다면 각각의 자원이 각기 다른 보안을 요구할 때보다 쉽게 정의할 수 있다.

자원에 대한 접근을 필요로 하는 역할(role)을 리스트 한다.

접근 제어(access control) 요구 사항으로 자원에 대한 접근을 필요로 하는지 아닌지를 설정하고, 만일 자원에 대한 접근이 필요하다면 어떤 종류의 접근이 필요한지를 결정한다. 동일한 자원에 접근하는 2가지 역할은 자원에 대한 동일한 접근을 필요로 하지 않을 수도 있다.

V.결론 및 향후 연구방향

웹서비스의 채택 및 지원은 현재 빠른 속도로 증가하고 있으며, 국내에서도 향후 2~3년 내에 가장 유망한 e-business 프레임워크로 전망되고 있다 [10]. 하지만, 전자상거래에 있어 보안 취약점을 가지고 있고 이를 극복하지 못하면, 아예 business 자체가 성립될 수 없음으로 보안 취약점에 대한 효과적인 대응책이 그 무엇보다 중요하다고 할 수 있다.

이에 본 논문에서는 웹서비스의 보안 강화를 위해 XML 보안 기술을 현재 적용 가능하거나 향후 개발 및 적용될 XML 기반의 보안 기술을 살펴봄으로써 웹서비스 프레임워크에서 제안하는 신뢰성 있는 사업 지원 방안을 논의하였다.

향후 연구로는 분석된 요구 사항을 만족하는 실제 웹서비스 시스템의 설계 및 구현을 통해 제안된 기법을 검증함으로써 추가적인 보안 취약 요소의 식별 및 대책에 대한 연구가 요구된다.

6. 참고 문헌

[1] WebService,
http://www.w3c.org/2002/ws/
[2] Simple Object Access Protocol (SOAP),
http://www.w3.org/TR/SOAP/
[3] WSDL,
http://www.w3.org/TR/wsdl/
[4] UDDI,
http://www.oasis-open.org/committees/uddi-spec/
[5] XML Signature Press Release
http://www.w3.org/2002/02/xmlsignature-pressrelease.html.en
[6] XML Encryption
http://www.w3.org/Encryption/2001/
[7] XML Key Management Specification
http://www.w3.org/2001/XKMS/
[8] Security Assertion Markup Language
http://www.oasis-open.org/committees/security/
[9] XML Access Control Markup Language
http://www.oasis-open.org/committees/xacml/index.shtml
[10] SOAP Security Extensions: Digital Signature
http://www.w3.org/TR/SOAP-dsig/