WS-Security를 적용한 서비스 공유 시스템 설계

왕현정°, 오인원°°, 윤혁중°°°, 최석진°°°, 박현동°, 류재철°

° 충남대학교 전기정보통신공학부 °° 웨어플러스 °°° 국가보안기술연구소

Design of service sharing system using WS-Security

Hyun-Jung Wang°, In-Won Oh°°, Hyuk-Joong Yoon°°°, Seok_Jin Choi°°°,

Hyun_Dong Park°, Jae-Cheol Ryou°

° Division of Electrical and Computer Engineering Chungnam National Univ., °° Wareplus, °°° National Security Research Institute

요 약

웹 서비스(Web Services)는 XML(eXtensible Markup Language)에 기반을 두고 있어 이기종 시스템간의 자연스러운 데이터 공유와 통신을 가능하게 한다. 또한 웹 서비스 기술은 서비스 제공자가 만든 서비스를 proxy object의 호출을 통해 필요한 곳에서 시스템 환경에 구애받지 않고 사용할 수 있게 한다는 데에 큰 의미가 있다. 웹 서비스 기술을 기반으로 구축된 시스템은 프로그램의 재사용성을 높일 수 있고, 이기종 시스템간의 상호운영성을 좋게 하며 사용자는 보다 양질의 서비스를 제공받을 수 있게 한다. 웹 서비스의 이러한 특성으로 인해 웹 서비스를 기반으로 하는 시스템에 대한 연구가 활발히 진행되고 있다. 그러나, 아직 웹 서비스 기술과 보안기술을 적용한 시스템에 대한 개발 사례가 미흡하다. 본 논문은 웹 서비스의 특징과 장점을 부각시킨 웹 서비스 기반의 서비스 공유 시스템을 제안하고, WS-Security 기술을 적용해 안전한 서비스 공유 시스템이 되도록 설계하고 안전성과 효율성을 분석한다.

1. 서론

웹 서비스의 근간을 이루는 표준 프로토콜인 SOAP(Simple Object Access Protocol), WSDL(Web Services Description Language), UDDI(Universal Description Discovery & Integration)는 모두 XML에 바탕을 두고 있다. XML은 데이터의 공개표준이다. HTML(Hyper Text Markup Language)은 정보의 내용을 표현하는 데에 국한된 기능만을 제공하지만, XML은 각 정보들의 관계를 표현할 수 있기 때문에 HTML과 비교할 때에 융통성이 좋은 기능을 제공할 수 있다. 이러한 특징을 가진 XML을 기반으로 하는 웹 서비스 기술은 이기종 시스템간의 데이터 공유를 가능하게 하고 서비스 제공자로부터 서비스를 필요로 하는 곳에서 개발환경이 시스템 환경에 구애받지 않고 사용할 수 있게 한다. 웹 서비스 기술의 간결성과 확장성은 기존 환경의 상당 부분을 개선하고 사용자들에게 더 많은 사용상의 편리함을 제공할 것이라는 예측도 있으나, 더 많은 보안상의 취약성을 가질 수 있다. 앞으로 웹 서비스를 기반으로 하는 시스템의 개발이 점차 활발해질 것으로 전망되며 웹 서비스를 기반으로 하는 시스템을 안전하게 구축하기 위해서는 XML에 대한 보안기능과 웹 서비스의 확장성을 뒷받침할 수 있는 보안기술이 요구되고 있다.

WWW(World Wide Web)기반 시스템에서도 메시지 무결성, 기밀성, 사용자인증의 보안기능을 필요로 한다. W3C에서는 XML 문서와 XML 형태로 이루어진 메시지에 대한 보안요구사항을 만족시키기 위해 XML Signature[1]와 XML Encryption[2] 표준 스펙을 발표하였고, OASIS에서는 WS-Security[3]를 발표하였다. XML Signature는 메시지의 무결성, 사용자인증 기능을 제공하며, XML Encryption은 메시지에 대한 기밀성을 제공한다. WS-Security에서는 XML Signature, XML Encryption

스펙을 포함하고, 보안토큰 전송에 대한 내용을 규정하고 있다[4].

본 논문에서는 기업 내에 물리적으로 분리되어 있고, 타 부서원은 사용할 수 없었던 부서별 관리 시스템을 웹 서비스 기반으로 하여 서비스를 공유할 수 있는 모델로 제안하고, 안전한 서비스 공유 시스템 구축을 위해 WS-Security 기술을 적용한다. 본 논문의 구성은 다음과 같다. 2장에서 웹 서비스 보안기술을 살펴보고, 3장에서는 서비스 공유 시스템을 설계하고 안전성과 효율성을 분석한다. 4장에서는 결론 및 활용방안을 서술한다.

2. 관련 연구

◇ WS-Security

전송되는 SOAP 메시지에 무결성, 기밀성을 제공하고 송신자 인증을 보장하는 방법과 인증서와 같은 보안토큰(Security Token)을 전파하는 방법을 규정하고 있다. WS-Security는 2002년 4월에 Version 1.0이 발표되었고, 같은 해 8월에 Version 1.0의 addendum이 발표되었다. MS, IBM과 Verisign이 협력하여 작성한 후에 이를 OASIS에 제출하였고, Sun은 OASIS를 통해 규격의 개선 작업에 참여하고 있다. XML-Signature, XML-Encryption 등 기존의 관련 표준을 모두 수용함으로써 기존 환경에 변화를 주지 않는다는 것이 특징이다. WS-Security에서 지원하는 기능은 다음의 세 가지이다[3].

○ 보안토큰 전파

WS-Security 규격의 주요 목적 중 하나는 보안토큰을 SOAP 메시지를 이용하여 다른 노드에 전달하는 것이다. 보안토큰이란 사용자의 신원을 증명할 수 있는 Claims의 집합을 의미한다. 보안토큰을 다루는 방법으로는 3 개의 element를 사용하는 방법이 있다.

- Username Token - 사용자의 ID와 패스워드를 전송하는 방식이다. 가장 기초적인 사용자 인증 방법으로 패스워드가 평문으로 전송되어 취약성을 내포하고 있지만, SSL 등과 같은 전송계층에서의 보안기술이 사용되는 환경에서는 가장 간단하게 사용할 수 있는 방법이다. addendum에서 replay attack을 방지하기 위해 <Nonce>와 <Created> 태그가 포함되었다.
- BinarySecurityToken - 이 방법을 사용하면 사용자가 자신의 X.509 인증서나 Kerberos에서 사용하는 티켓을 포함하여 전송할 수 있다.
- SecurityTokenReference - 이 헤더에는 실제로 SecurityToken을 포함시키지 않고, 다른 헤더에 있는 인증정보를 참조하거나 또는 특정한 사이트에 존재하는 정보를 참조하도록 한다. [그림 1]을 보면, www.XYZShoes.com 사이트의 특정 위치에 있는 X.509 형식의 인증서를 참조하도록 하고 있다.

```
<wsse:SecurityTokenReference
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2-02/04/secext">
    <wsse:Reference
      URI="http://www.XYZShoes.com/tokens/XYZ#X509token">
</wsse:SecurityTokenReference>
```

[그림 1] SecurityTokenReference의 예제

○ 메시지 무결성

WS-Security에서 메시지의 무결성은 XML Signature 기술에 바탕을 두고 있다. SOAP 메시지 내부의 특정 부분을 지정하여 이에 송신자의 전자서명을 계산하여 전송하는 방법으로 메시지의 무결성을 보장한다. 대상이 되는 부분의 해쉬함수 계산결과와 이를 송신자의 비밀키를 이용하여 계산한 전자서명 결과를 함께 포함한다. 이를 검증하기 위해 필요한 공개키 등의 정보는 앞에서 설명한 보안토큰을 이용하여 전송한다[5][6].

○ 메시지 기밀성

WS-Security에서 메시지의 기밀성은 XML Encryption 기술에 바탕을 두고 있다. SOAP 메시지의 <Header> 부분에 포함되는 내용의 전체 또는 일부를 특정 수신자만이 복호화 할 수 있도록 암호화한다. 먼저, 대상이 되는 부분을 대칭키 암호방식을 이용하여 암호화하고 사용된 대칭키를 수신자의 공개키로 암호화하여 기밀성을 보장한다. 만약, 하나의 메시지에 각 수신자별로 열람할 수 있는 부분이 따로 존재한다면 송신자는 수신자별로 데이터를 구분하고 이를 별도의 대칭키를 이용하여 암호화함으로써 수신자별 기밀성을 제공할 수 있다[5][6].

3. 서비스 공유 시스템 설계

◇ 서비스 공유시스템 구축 필요성

기존 분산처리 환경에서의 이기종 시스템간 통신은 양단간에 시스템을 구성하고 있는 운영체제와 프로그램을 작성한 언어에 영향을 받았으며 각 시스템에서 구현한 프로그램을 재사용하기 힘들었다. 분산처리를 위해 CORBA나 RMI 같은 기술들이 등장하였으나, 그러한 기술들은 대부분 특정 포트를 통해 통신이 이루어지는데, 대부분의 기업은 방화벽을 사용하고 있고, 이 방화벽에 막혀서 원활한 통신이 이루어지지 않는 경우도 발생할 수 있었다. 기존 시스템은 이러

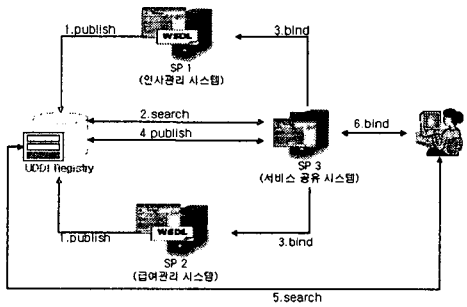
한 문제점들을 안고 있었기에 이기종 시스템간의 상호운영성이 좋지 않았다. 그러나 웹 서비스는 데이터 표준으로 자리 잡은 XML을 기반으로 한 표준 메시징 방식을 사용하므로 구현된 언어나 운영체제, 시스템 환경에 관계없이 서로 메시지를 교환할 수 있게 되었다. 또한 SOAP은 기존 통신 프로토콜과 바인딩하여 메시지를 전송하므로 보안을 위해 설치된 방화벽이 있을 때에도 만약 HTTP 프로토콜과 바인딩하여 메시지를 전송할 경우 방화벽에 막히지 않고 통신할 수 있는 장점이 부각되고 있다. 또한 서비스 제공자가 자신이 만든 응용프로그램을 개인의 PC에 설치하는 방식이 아니고, 웹을 통해 필요한 서비스 사용자에게 제공하고, 프로그램의 관리는 서비스 제공자가 맡아서 처리하므로 사용자는 내부적으로 버전이 변경되거나 프로그램에 수정이 이루어져도 다시 프로그램을 설치하거나 자신의 시스템을 변경하는 번거로움 없이 서비스를 사용할 수 있다.

◇ 서비스 공유 시스템 설계

○ 서비스 공유 시스템 전체 구조도

[그림 2]는 본 논문에서 제시하려는 서비스 공유 시스템을 웹 서비스의 전체 관점에서 봤을 때의 모습이다. [그림 2]에서 서비스 제공자 SP(Service Provider)1, SP2가 제공하는 웹 서비스를 SP3가 서비스 요청자가 되어 바인딩을 통해 서비스를 사용하며, SP1, SP2에서 제공하는 서비스를 통합하여 새로운 서비스 공유 시스템을 구축하고 SP3가 다시 웹 서비스 제공자로서의 역할을 하는 모습이다.

서비스 제공자 SP1, SP2는 B2B환경에서는 각기 다른 기업이 될 수 있고, 기업 내의 환경이라면 각기 다른 부서 또는 각기 다른 응용프로그램이라고 생각할 수 있다.



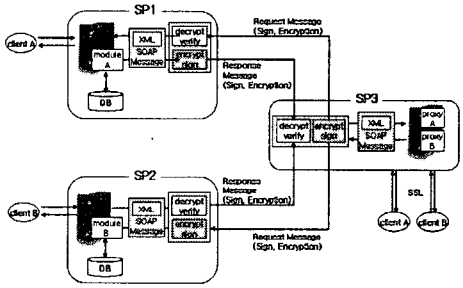
[그림 2] 서비스 공유 시스템 구조도

시스템별 역할은 다음과 같다.

- SP1 - 인사관리 부서의 인사관리 시스템이며 사원에 대한 정보를 저장하고 관리한다.
- SP2 - 급여관리 부서의 급여관리 시스템이며 사원의 급여정보를 저장하고 관리한다.
- SP3 - 인사관리 시스템과 급여관리 시스템의 웹 서비스 모듈을 사용하여 새로운 서비스를 제공하는 서비스 통합 시스템이다.

○ WS-Security를 적용한 서비스 공유 시스템 구조도

[그림 2]에 보안기술을 적용시킨 서비스 공유 시스템은 [그림 3]과 같다. [그림 3]에서는 UDDI에 서비스를 등록하고, 검색하는 부분은 생략하였고, 서비스 제공자와 서비스 요청자의 상호 작용에 중점을 두었다. SP3는 웹 서비스 기술을 이용해 구현된 SP1의 서비스와 SP2의 서비스를 호출해서 사용한다. 내부적으로는 SP1과 SP2에서 제공하는 서비스를 사용하기 위한 Proxy Object를 생성하고, Proxy Object의 메소드를 호출함으로써 SP1, SP2의 서비스를 사용한다. 클라이언트 A와 클라이언트 B는 이제 SP3에서도 SP1이나 SP2에서와 같은 서비스를 제공받을 수 있다.



[그림 3] 보안모듈을 추가한 서비스 공유 시스템 구조도

SP1, SP2와 SP3간에 메시지 교환이 일어날 때 데이터의 안전한 교환을 위해 보안기능이 요구된다. 메시지의 무결성과 사용자 인증을 위해 XML Signature를 사용하고 메시지에 대한 기밀성을 제공하기 위해 XML Encryption을 적용한다. 본 설계에서는 클라이언트가 SP3에 접속할 때 아이디/패스워드 기반의 인증방식을 사용하며, 이 과정에서 해킹방지를 위해 SSL을 적용한다. 본 논문에서는 서비스 공유에 목표를 두었기에 클라이언트의 인증부분을 기존의 Basic Authentication 방식으로 구성한다.

◇ 안전성/효율성 분석

본 논문에서 설계한 서비스 공유 시스템은 각 SP 사이에서 전송되는 정보에 대해 기밀성과 전자서명을 제공한다. 이러한 기능은 기존의 SSL을 이용해서도 어느 부분까지는 제공해 줄 수 있다. 그러나, SSL을 사용하는 방법에서 제공하지 못 하는 몇 가지 장점이 있는 데, 이를 살펴 보면 다음과 같다.

- 기밀성 유지 : 통신 당사자 사이에 중간 매개자가 있는 환경에서 SSL을 사용하게 되면 중간 매개자들이 전송되는 정보의 내용을 복호화하여 다시 암호화하므로 그 내

용을 합법적으로 열람할 수 있다. 본 논문에서는 최초 정보 제공자와 사용자 사이에 하나의 매개자만이 있으므로 이러한 취약성이 부각되지 않았으나, SP3와 사용자 사이에 또 하나의 SP가 위치한다면 SP3에서 정보가 노출된다. 그러나, XML 보안기법을 사용하게 되면 송신자가 특정 수신자만을 위한 암호화를 수행할 수 있으므로 중간 매개자에서 정보가 노출되는 상황을 방지할 수 있다.

- 서비스 통합의 간편 : 기존 환경에서 두개 이상의 별도 서비스를 하나의 프로그램에서 지원하기 위해서는 통합 서비스 프로그램을 별도로 제작해야 한다. 그리고, 별도 서비스가 업그레이드되면 통합한 프로그램도 업그레이드를 수행해야 하는 부담이 있다. 그러나, 웹 서비스를 사용하면 별도 서비스가 업그레이드된다 하더라도 통합 서비스 모듈에서는 단순히 업그레이드된 별도 서비스를 호출하면 되므로 버전 교체 등의 부담을 덜 수 있다.
- 암호 대상의 부담 감소 : SSL은 양단에서 전송되는 모든 정보에 대해 암호화를 수행하므로 암호화가 필요하지 않은 부분에 대해서도 불필요한 암호화를 수행하게 된다. 그러나, XML 보안기법을 사용하게 되면 실질적으로 암호화가 필요한 데이터 부분에 대해서만 선별적으로 암호화를 수행할 수 있어서 암호화에 대한 부담을 감소시킬 수 있다.
- PKI와 연동 : 본 논문에서는 구현되지 않았으나 XKMS 서버를 설치하면 X.509 이외의 다양한 인증서도 사용할 수 있고, XML 문서 내에서 인증서와 관련된 동작을 간단히 처리할 수 있다. 이는 지정된 형태의 인증서만을 사용해야 하는 SSL에 비해 프로그램의 구성을 융통성을 높일 수 있다.

4. 결론 및 활용방안

본 논문에서는 기존 분산처리 시스템의 단점을 해결하고자 웹 서비스를 기반으로 한 서비스 공유 시스템을 제안하여 기존 시스템보다 프로그램의 재사용성과 상호운영성을 높일 수 있고, 개발자의 입장에서는 개발 속도를 향상시키고 보다 비즈니스 로직에 집중할 수 있게 해줄 수 있는 시스템을 설계하였다. 또한 웹 서비스를 기반으로 한 시스템에 적합한 보안기능을 부여하기 위해 기존에 사용되던 SSL 뿐만 아니라 SOAP 메시지에 부분적인 암호화를 가능하게 하는 XML Encryption과 전자서명을 가능하게 하는 XML Signature 기술을 내포하고 있는 WS-Security를 적용하였다. WS-Security는 서비스 공유 시스템에 교환되는 메시지에 무결성, 기밀성, 사용자인증 기능을 제공하여 교환되는 데이터와 사용하는 서비스 모두에 안정성을 제공하였다. 그러므로 사용자는 시스템에 대한 신뢰를 가지고, 필요한 서비스를 사용할 수 있게 된다.

본 논문에서는 기업 내의 서비스 공유를 목표로 하였지만 범위를 넓혀 기업 간 시스템 통합에도 이용할 수 있으리라 본다. 웹 서비스는 기존의 IT 환경에 큰 폭의 변화을 가져올 수 있는 기술이다. 특히 다른 서버에서 작동하고 있는 서비스들을 통합하여 사용자가 하나의 서버에 접속하여 원하는 모든 프로그램의 서비스를 받을 수 있다는 점은 수많은 시스템을 운영하고 있는 기업의 입장에서 볼 때에 업무의 효율성 제고와 하드웨어 및 소프트웨어 유지비용의 절감이라는 이득을 얻을 수 있게 된다. 본 논문에서는 기업 내의 서비스 공유 시스템을 설계하였지만, 향후에는 더욱 큰 규모의 서비스 공유, 여러 기업들 간의 서비스 통합을 위한 연구가 진행되어야 할 것이다.

[참고문헌]

[1] W3C, XML-Signature Syntax and Processing, http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/, 2002

[2] W3C, XML Encryption Syntax and Processing, http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/, 2002

[3] OASIS, Web Services Security : SOAP Message Security, http://www.oasis-open.org/committees/download.php/3281/WSS-SOAPMessageSecurity-17-082703-merged.pdf, 2003

[4] Bret Hartman, Donald J. Flinn, Mastering Web Services Security, Wiley, 2003

[5] Ben Galbraith, Whitney Hankison, Professional Web Services Security, Wrox, 2002

[6] Blake Dournaee, XML Security, RSA Press, 2002