웹 서비스를 이용한 통합 승인 시스템에 관한 연구

한성근, 김규석, 김주영
한국과학기술정보연구원
e-mail:{sghan,gskim,jyghim}@kisti.re.kr

A Study on Integrated Approval System using Web Services

Sung-Geun Han, Gyu-Seok Kim, Joo-Yeong Kim
Korea Institute of Science and Technology Information

요 약

부서마다 하드웨어 및 시스템 자원을 독립적으로 관리하는 조직에서 보안 관리와 같은 일부 중요한 기능을 전체적으로 통제하고 관리하기 위해서는 각 부서에서 운영하는 시스템과 연동하는 통합 시스템이 필요하다. 본 논문에서는 웹 서비스를 활용하여 분산된 각 부서의 웹 시스템에 공용 서비스를 제공 및 관리함으로써, 각 부서의 특화된 시스템에 영향을 최소화하고 주요 기능은 전체적으로 관리 통제할 수 있는 시스템 설계에 대해 다룬다.

1. 서론

최근 수년간 인터넷 사용의 폭발적 증가, 네트워크 기술의 급속한 발전과 같은 소식은 방송 매체를 통해 수없이 듣고 있는 상황이다. 특히, 국내에서는 인터넷을 떠나서는 업무를 제대로 할 수 없을 정도로 인터넷 특히 웹 시스템은 우리 생활과 밀접한 관계를 가지고 있으며, 웹 2.0 시대, 3.0 시대로 발전되어 가고 있다. 이미 기업과 같은 업무 조직에서도 웹 환경의 업무 시스템으로 전환되어 가고 있으며, 더욱더 편리한 웹 인터페이스 환경이 제공되고 있다.

하나의 조직 내에서 부서마다 독립된 업무를 수행하는 경우, 일반적으로 개별 부서들은 부서 차원의 웹 환경을 구축하게 되고 이에 소요되는 하드웨어 및 기타 필요한 자원들을 자체적으로 관리하게 된다. 각 부서에 속한 부서원들의 자원 요청이 있을 시 개별적으로 운영되고 있는 승인 시스템을 사용하여 인가해준다. 각 부서의 승인 시스템은 각 부서마다 특화되어 있으며, 다른 부서에서는 사용할 수 없거나 복잡한 과정을 거치도록 되어 있다.

최근 인터넷 사용과 더불어 보안 문제가 큰 이슈가 되고 있으며, 조직 내에서도 총괄 보안 부서를 따로 두고 전체 조직을 관리하도록 하고 있다. 이에 따라 개별 부서들 모두 해당 부서의 자원을 사용한다고 하더라도 총괄 보안 부서에서 정한 방식의 프로세스를 따르도록 요구되고 있다. 이럴 경우, 각 개별 부서에서 운영하고 있는 승인 시스템은 모두 총괄 보안 부서에서 정한 프로세스대로 다시 수정되어야 하며, 개별 부서가 많을 경우 이에 따른 비용이 만만치 않게 된다. 또한, 신규 개별 부서가 만들어질 경우에도 총괄 보안 부서에서 정한 프로세스를 그대로 적용한 승인 시스템을 중복적으로 만들어 나가야 하는 문제가 발생한다.

따라서, 본 논문에서는 웹 서비스를 활용하여 통합 승인 시스템(Integrated Approval System)을 설계함으로써 총괄 보안 부서에서 정한 승인 프로세스 개발에 대한 중복을 피하고, 개별 부서에서 쉽게 연동할 수 있는 시스템을 구축할 수 있는 공통 모듈에 대해 다룬다.

2. 관련 연구

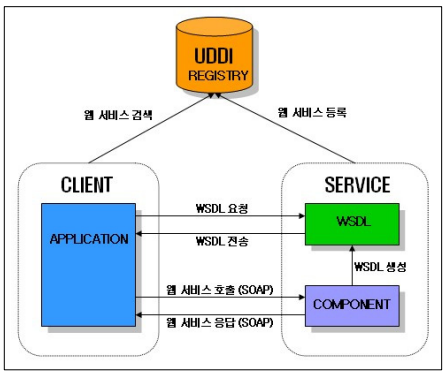
2.1 웹 서비스 (Web Services)

웹 서비스는 HTTP나 SMTP와 같은 표준 기반 인터넷 프로토콜 상에서 XML을 이용하여 서로 다른 운영 체제의 서로 다른 하드웨어 플랫폼 상의 소프트웨어 컴포넌트를 액세스할 수 있는 기술이다. 즉, 기업 내·외부에서 웹을 통해 제공되는 각 기능 단위의 애플리케이션을 연동해 새로운 가상 서비스를 제공하도록 하는 IT 기술이다. 모든 종류의 정보시스템을 통합·연계 및 정보시스템 내에 존재하는 문서, 데이터베이스, 응용 소프트웨어들을 공동 사용할 수 있도록 지원한다. CORBA나 DCOM 또한 이기종간의 호환성을 목표로 하고 있지만, 벤더 중심의 프로토콜을 사용함으로써 호환성 문제를 해결하지 못하였다. 인터넷을 통한 클라이언트와 서버 간 통신은 수많은 장벽

에 직면하게 되는데, 보안 및 네트워크 담당자들은 인터넷에 전달되는 수많은 잘못된 정보들을 차단하기 위해 라우터 및 방화벽을 설치하고 운영 관리한다. 이럴 경우 DCOM, CORBA, Java RMI에서 사용하는 독자적인 프로토콜은 인터넷 환경에서 적합하지 않다.

웹 서비스는 다음과 같은 특징을 가진다.

- HTTP 프로토콜 이용 : 방화벽에 막히지 않고 서비스를 호출할 수 있다.
- XML 기반 : XML을 이용해서 메시지를 전달하고 응답받기 때문에 플랫폼과 구현 언어에 독립적이다.
- 느슨한 결합 (Loosely Coupled) : 클라이언트와 서버 로직이 느슨하게 결합되어 서로 다른 시스템 사이의 통합을 단순화시킬 수 있다.
- 동기식 혹은 비동기식 운영 가능성 : 기존의 분산 컴포넌트 모델들은 RPC 형태의 동기적인 서비스만이 가능했지만, 웹 서비스는 단지 메시지만 보내는 비동기적인 형태로도 이용 가능하다.
- 기존의 시스템에 적용 가능 : 웹 서비스를 제공하는 시스템을 구현하기 위해서 기존의 시스템을 새롭게 구현하는 것이 아니라, 웹 서비스 통신을 할 수 있는 컴포넌트만 추가함으로써 웹 서비스를 제공하는 시스템을 구현할 수 있다.



(그림 1) 웹 서비스 상호 작용

(그림 1)은 웹 서비스를 사용한 시스템 구축 시 클라이언트와 서버(서비스) 간의 상호 작용을 나타낸다. 서비스 제공자는 웹 서비스 이용 방법에 대한 WSDL(Web Services Description Language) 문서를 XML로 작성하고, UDDI(Universal Description, Discovery, and Integration)에 웹 서비스 정보를 등록한다. 서비스 이용자는 UDDI를 통해 WSDL 문서의 URL을 검색한 후, WSDL 문서를 얻어 웹 서비스 이용 방법을 분석한다. 서비스 이용자는 SOAP 메시지를 생성해서 HTTP 프로토콜 상에서 웹 서비스를 호출함으로써 서비스 제공자의 서비스를 이용할 수 있다. 이와 같이 웹 서비스는 클라이언트/서버 간 느슨한 결합을 통해 독립된 개별 시스템들을 서비스 중심 서버에 쉽게 연동할 수 있는 기반을 마련하고 있다.

2.2 설계시 요구 사항

본 논문에서는 다음과 같은 상황을 가정한다. 하나의 조직에 개별적인 부서가 존재하며, 각 부서는 독립된 자원 관리 승인 시스템을 사용하고 있다. 그러나, 각 부서의 승인 시스템에서 제공하는 서비스 중 보안 사항에 관한 서비스는 새롭게 정해진 총괄 보안 부서의 승인 프로세스를 따라야 한다는 과제가 주어진 상황이다.

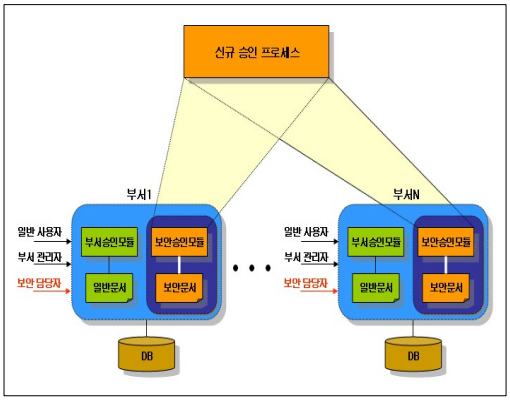
위와 같은 상황을 해결하기 위해서는 다음과 같은 요구 사항이 필요하다.

- 유지 비용 최소화 : 새로운 기능의 프로세스가 만들어져서 적용될 경우 각 개별 부서의 승인 시스템의 수정은 불가피하다. 개별 부서가 많을 경우 그 만큼 작업량이 증가하므로 수정할 내용이 최소화되어야 한다. 또한, 총괄 보안 부서의 승인 프로세스 자체가 수정되었을 경우, 쉽게 그 내용을 반영할 수 있도록 해야 한다.
- 다양한 개발 플랫폼 지원 : 새롭게 적용될 프로세스는 소프트웨어나 하드웨어 독립적으로 구현 가능하도록 하여 다양한 플랫폼에서 개발 및 적용 가능하도록 해야 한다.
- 로그인 최소화 : 기존의 개별 부서 승인 시스템은 해당 부서 내에서 관리자 혹은 부서장의 승인을 받으면 된다. 그러나, 총괄 보안 부서의 승인 프로세스가 추가되었을 경우 상위 부서의 보안 담당자의 승인을 받을 필요가 있다. 그럴 경우 상위 보안 담당자는 각 개별 부서의 승인 시스템에 일일이 로그인하여 승인 처리를 해 주어야 한다. 개별 승인 시스템이 많을 경우 상위 보안 담당자는 모든 시스템을 모니터링하고 있어야 하는 문제가 발생한다. 따라서, 상위 보안 담당자가 쉽게 해당 부서의 승인 요청을 처리할 수 있도록 시스템을 구성해야 한다.

3. 시스템 설계

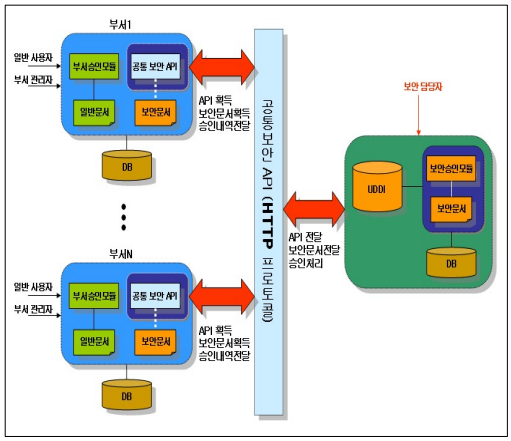
기존 승인 시스템에 새로운 기능을 넣는 일은 단순히 새로운 모듈을 개발하여 추가하는 작업일 수도 있다. (그림 2)는 신규 승인 프로세스에 적합한 모듈을 개발하여 각 부서마다 사용하고 있는 승인 시스템에 추가한 그림을 나타낸 것이다. 각 부서는 새롭게 추가된 정책을 반영하기 위해 부서의 문서들을 일반 문서와 보안 담당자의 승인을 거쳐야하는 보안 문서를 분류할 필요가 있다. 일반 문서는 기존 방식대로 해당 부서의 승인만 거치면 되며, 보안 문서는 보안 담당자의 승인을 거친다. 보안 담당자는 상위 부서인 총괄 보안 부서의 담당자를 의미한다. 이와 같은

시스템은 다음과 같은 문제점을 가진다. 첫째, 총괄 보안 부서의 보안 담당자가 승인을 하기 위해서는 각 부서에서 제공하는 승인 시스템에 로그인하여 해당 승인 처리를 수행한다. 개별 부서가 많을수록 로그인에 대한 부담이 증가할 것이다. 개별 부서마다의 로그인을 피하기 위해 SSO(Single Sign-ON)를 구현한다고 할지라도 개별 부서의 승인 처리를 위해서는 개별 부서의 승인 시스템에 접근해야하는 불편은 막을 수 없다. 둘째, 총괄 보안 부서에서 승인 프로세스를 수정하였을 경우 이를 반영하기 위해서 각 부서의 보안 승인 모듈을 모두 수정해야 한다. 즉, 각 부서의 승인 시스템이 보안 승인 모듈에 밀결합(tightly-coupled)되어 있어 승인 프로세스를 변경하고 일관성을 유지하기 위해 많은 어려움이 따른다.



(그림 2) 개별 시스템에 승인 모듈 추가

이러한 문제점을 해결하기 위해 본 논문에서는 보안 승인 모듈을 공통 모듈로 개발하고, 해당 모듈을 사용하기 위한 규약(공통 보안 API)을 제시하여 각 부서의 개별 승인 시스템으로 하여금 공개된 API 만을 사용함으로써 신규 승인 프로세스를 적용받을 수 있는 시스템을 개발할 수 있도록 한다. (그림 3)은 웹 서비스를 사용하여 보안 문서의 승인을 처리할 수 있는 통합 승인 시스템을 나타낸다. 여기에서도 새로운 보안 승인 정책을 따르기 위해 먼저 일반 문서와 보안 문서를 분류할 필요가 있다. 일반 문서에 대한 승인은 개별 부서의 승인 시스템에서 구현한 모듈에 기반하여 처리되지만, 보안 문서의 경우 공통 보안 API를 통해 통합 승인 시스템에 전달된다. 보안 담당자는 개별 승인 시스템에 로그인 할 필요 없이 통합 승인 시스템에 접속하여 각 부서에서 요구된 승인 처리를 수행하면 된다.



(그림 3) 웹 서비스를 사용한 통합 승인 시스템

통합 승인 시스템에서는 보안 승인 모듈을 만들고 모든 부서의 개별 시스템과 연동하기 위해 웹 서비스를 이용하여 공통 보안 API를 등록한다. 총괄 보안 부서에서 보안 프로세스가 변경되었을 경우 통합 승인 시스템의 보안 승인 모듈만 수정하면 되므로 부서의 개별 승인 시스템에 영향을 끼치지 않는다. 또한 통합 승인 시스템에서는 보안 문서에 대한 파일을 XML로 제공하여 개별 시스템에서 활용할 수 있도록 함으로써 전체 시스템의 일관성을 유지할 수 있다.

3.1 보안 문서 정의

보안 승인을 위해서는 총괄 보안 부서에서 보안 문서를 정의할 필요가 있다. 다음은 보안 문서를 작성 주기에 따라 분류한 예이다.

- 정기 문서 : 로그관리대장, 퇴실자보안점검표, 보조기억매체 점검대장, 무선랜 사용대장, 시스템계정 관리대장, 백업관리대장 등.
- 비정기 문서 : 보조기억매체 관리대장, 사용자 계정 관리대장, 장애처리일지, 비밀취급인가 발급대장, 폐기관리대장, 외부접근/허가 신청서, 변경작업 요청서, 침해사고 발생신고서, 보안성 검토 요청서 등.

위와 같은 문서를 XML을 사용하여 작성 및 배포함으로써 플랫폼에 관계없이 일관된 서비스 구현이 가능하다. 다음은 사용자 계정 신청서에 대한 XML 문서의 예이다.

```
<?xml version="1.0" encoding="euc-kr" standalone="no"?>
<!DOCTYPE userlist SYTEM "ACCOUNTList.dtd">
<accountlist>
    <account>
        <applicant>
            <name>한성근</name>
            <dep>정보시스템운영팀</dep>
            <date>20080918</date>
        </applicant>
```

```
        <purpose>웹 페이지 개발</purpose>
        <access>
            <hostname>myhost</hostname>
            <ip>192.168.113.111</ip>
        </access>
        <userid>myname</userid>
        <userpass>password</userpass>
        <notes>메모</notes>
    </account>
</accountlist>
```

3.2 공통 보안 API 정의

웹 서비스를 통해 개별 부서의 승인 시스템은 공통 보안 API를 사용함으로써 플랫폼에 상관없이 연동할 수 있다. 주요 API는 <표 1>과 같다

<표 1> 공통 보안 API

API 명	INPUT	OUTPUT	설명
ReqApproval	DID - 보안문서 신청 번호 FormURL - 작성된 문서의 디스플레이 URL	1 - SUCCESS 0 - FAIL	새로운 보안 문서 신청서를 작성하고 통합승인시스템에 승인을 요청한다.
GetStatus	DID - 보안문서 신청 번호	1 - 승인완료 0 - 처리 중 -1 - 반려	승인 신청한 문서에 대한 현재 처리 상태 정보를 얻는다.

웹 서비스에서 플랫폼 독립적으로 서비스를 호출하고 응답하기 위해 XML 메시징 프로토콜인 SOAP(Simple Object Access Protocol)을 사용한다. 물론, SOAP은 앞에서 언급했듯이 HTTP 프로토콜을 사용한다. 서비스 이용자는 UDDI를 통해 WSDL 문서를 얻어서 분석한 후 SOAP 메시지를 생성하여 웹 서비스를 호출하고, 응답된 SOAP 메시지를 분석하여 처리한다. 다음은 각 단계별 처리 메시지 정보를 나타낸다.

- WSDL을 통해 얻은 API(메소드) 정보

```
int ReqApproval(String DID, String FormURL);
```

- 웹 서비스 호출을 위한 SOAP 메시지

```
<ReqApproval>
    <String_1 type="xsd:string">DID0001</String_1>
    <String_2    type="xsd:string">http://localhost/view.jsp
    </String_2>
</ReqApproval>
```

- 웹 서비스의 응답 SOAP 메시지

```
<ResponseReqApproval>
    <result type="xsd:int">1</result>
</ResponseReqApproval>
```

4. 결론 및 향후 계획

부서마다 개별적인 승인 시스템을 사용하고 있는 하나의 조직에서 개별 시스템의 일부 승인 기능을 총괄 부서에서 통제하기 위해서 본 논문에서는 통합 승인 시스템을 설계하였다. 제시한 통합 승인 시스템은 웹 서비스를 기반으로 하여 플랫폼 독립적으로 개별 시스템과 연동 가능하며 총괄 부서의 정책을 일관적으로 유지할 수 있는 장점을 가진다. 향후 계획으로는 본 논문에서 설계한 내용을 바탕으로 실제 웹 서비스 기반의 통합 승인 시스템을 구현하는 것이다.

참고문헌

[1] Henry Bequet, Meeraj Moidoo Kunnumpurath, Sean Rhody, Andre Tost, "Beginning Java Web Services", WROX, 2003

[2] 유석환, 차무홍, 신동일, 신동규, "웹 서비스를 위한 통합접근 관리 연구", 한국정보과학회 가을 학술발표논문집, Vol. 31, No. 2, pp.349-351, 2004

[3] 김귀남, "XML 전자서명을 이용한 다중인증 멀티 에이전트시스템", 정보보증논문지, Vol. 5, No. 1, 2005.3

[4] 최진탁, "Single Sign-On을 이용한 인증 관리 기법에 관한 연구", KSIAM IT series, Vol. 10, No. 1, pp.61-69, 2006

[5] 서대희, 이임영, "멀티 에이전트를 이용한 Single Sign-On 인증 모델에 관한 연구", 한국통신학회논문지, Vol. 29, No. 7c, pp.997-1006, 2004

[6] 허미영, 현욱, 강신각, "폐쇄 사용자 그룹을 위한 정보공유 관리 시스템의 개발", 한국해양정보통신학회 추계종합학술대회지, Vol. 4, No. 2, pp.324-327, 2000

[7] 김기성, 김광, 허신, "이기종 시스템에서 안전한 데이터 전송을 보장하는 웹 보안 모듈의 설계 및 구현", 정보과학회논문지:소프트웨어 및 응용, Vol. 32, No. 12, pp.1238-1246, 2005