모바일 환경을 위한 웹 서비스의 무단절 보안 연결에 관한 연구*

김용태** · 정윤수*** · 박길철**

요 약

모바일 디바이스의 기능이 강화되며 이의 사용이 빈번해 지면서 다른 서버 기반 어플리케이션과 더욱 복잡한 상호 작용의 필요성이 요구된다. 그리고 모바일 환경은 이동성에 의하여 웹 서버와 클라이언트인 모바일 디바이스 사이에 순간 단절의 문제가 발생하여 웹 서비스 지속성에 문제를 발생하고, 기존의 인증된 보안에도 영향을 미쳐 재 인증의 문제를 야기한다. 따라서 본 논문은 HTTP 기반 웹 서비스와 모바일 디바이스의 상호 연결성 확보와 보안성을 유지하며 지속적인 웹 서비스를 위하여 모바일 웹 서비스의 표준 프로토콜 기반의 모바일 환경에서 WAP을 사용하는 모바일 디바이스의 웹 서비스 접근을 위한 프레임워크를 구현한다. 본 논문은 HTTP와 WAP의 접근을 기초를 둔 비교된 성능 지연, 데이터 전송 볼륨에 대하여 같은 기준의 조건에서 분석된다. 또한 상호 작용 프로세스가 가져오는 실행 오버헤드 증가를 조사한다.

A study on Seamless Security Connection of Web Services for Mobile Environment

Yong-Tae Kim** · Yoon-Su Jeong*** · Gil-Cheol Park**

ABSTRACT

Other server based-application and the need of more complex interaction is required with functional strengthening and variational uses of mobile device. And mobile environments make a problem of continuous of web-service by making a problem of instance cutting between web-server and mobile device which acts as a client because of mobility. Therefore, this paper embodies a framework which keeps security and connectivity between HTTP based web-service and mobile device, and is for connection to web-service of mobile device which uses WAP in a standard protocol based mobile environment of mobile web-service for continuous web-service. This paper is analyzed in the same standard condition to compared functional delay which is based on HTTP and WAP access, and data transmission volume. Also it investigates improvement of execution overhead which is brought by interaction process.

Key words : Web Service, Mobile Device, Web Service Security, SSL(Secure Socket Layer)

---

* 본 연구는 산업자원부 지역혁신센터 사업인 민군겸용 보안공학 연구센터 지원으로 수행되었음.

** 한남대학교 멀티미디어학부

*** 교신저자, 충북대학교 전자계산학과

1. 서 론

차세대 XML 기반의 프레임워크인 웹 서비스는 상호운용성에 의해 다양한 분야에서 활용되면서 인터넷에 대한 의존도를 증가시킨다[1]. 또한 다양한 웹 서비스의 동적인 발견과 합성으로, 부가가치를 가진 새로운 형태의 웹 서비스 생성이 가능해지면서 분산 환경의 웹 서비스 아키텍처에서 보안의 중요성은 점점 더 강조되고 있다.

특히 웹의 구조적인 특성과 접근 용이성에 의하여 웹 애플리케이션에 대한 침입 시도가 증가되기 때문에, 웹 서비스의 보안은 단순한 중요성 이상의 의미를 갖는다[2].

현재 웹 서비스 보안은 유선 환경에 집중되어 있으며, SSL(Secure Socket Layer) 기반의 인터넷 통신 보안, 전자 서명, 보안을 포함한 S-HTTP (Secure-HTTP) 그리고 암호화 기능을 포함하는 S/MIME(Secure-MIME) 프로토콜 등이 존재하지만 모바일 환경의 특성을 고려한 보안에 관한 연구는 부족한 실정이며, 최근에 유무선 통합 환경에 대한 관심이 점차 고조되고 있다.

모바일 환경은 이동성에 의하여 웹 서버와 클라이언트인 모바일 디바이스 사이에 순간 단절의 문제가 발생한다. 이러한 순간 단절은 지속적인 웹 서비스에 문제를 발생하고 기존의 인증된 보안에도 영향을 미치고 재 인증의 문제를 야기한다.

그러므로 본 논문은 모바일 환경을 고려한 연결성 확보와 보안성을 유지하며 지속적인 웹 서비스를 위하여 모바일 환경에서 로밍 구간, 음영 구간의 이동에서 발생하는 문제인 순간 단절을 해결한다. 또한 어떠한 단절 상황에도 인증된 보안 확보를 위하여 연결 세션 관리에 의한 보안성을 향상시키는 XML/SOAP 메시지 처리 기반의 연결 세션 관리를 위한 프레임워크를 설계한다.

이 논문의 구성은 다음과 같다. 제 2장은 관련 연구에 대하여 기술하고, 제 3장은 본 논문에서 제안한 무단절 모바일 웹 서버 프레임워크에 대한 아키텍처 설계를 나타내고, 제 4장은 제안 시스템의 구현 기술과 평가를 기술한다. 그리고 제 5장은 결론을 나타내고 미래의 작업 과정에 대하여 기술한다.

2. 관련 연구

2.1 웹 서비스의 정의와 특징

웹 서비스는 클라이언트와 서버 아키텍처 기반의 서비스 공급자와 소비자로 구성되며, 클라이언트가 서버에 접근하기 위해 전통적인 통신 프로토콜을 사용한다. 웹 서비스의 공통적인 접근 패턴은 요구와 응답으로 구성된다. 클라이언트는 서버에게 수행 오퍼레이션과 오퍼레이션 수행을 위한 관련된 정보를 명시하는 요구 메시지를 전송한다. 서버는 명시된 오퍼레이션을 수행하고 응답 메시지로 응답한다[3].

웹 서비스 구현을 위해서는 XML 기반의 SOAP (Simple Object Access Protocol), WSDL(Web Service Description Language), UDDI(Universal Discovery Description and Integration)와 같은 기술들이 사용된다. WSDL은 XML의 특별한 형태로 웹 서비스 인터페이스를 기술하고, 오퍼레이션, 데이터 형식, 바인딩 전송과 endpoints 즉 URL을 기술한다. 웹 서비스는 서비스 공급자에 의해 공표되고 서비스 요청자에 의해 위치된 UDD이며, 디렉토리 모델을 사용하며, 웹 서비스 공급자에 대한 정보를 포함한다. 표준 분류와 WSDL 파일을 통합에 의한 서비스의 인터페이스를 기술하는 범주를 포함한다.

웹 서비스 인스턴스는 SOAP 프로토콜을 통해 통신한다. XML(eXtensible Markup Language) 메시지 인코딩에 기반하고 HTTP와 같은 임의의 전송 프로토콜을 통해 전송되고, 그리고 클라이언트가 다른 어플리케이션 사이에 원격 프로시듀어 호출(RPCs) 수행을 담당한다[4].

2.2 모바일 디바이스 환경

모바일 디바이스는 상대적으로 낮은 처리 능력, 작은 메모리, 사용자 인터페이스 제한, 그리고 제한된 대역폭을 사용하므로 디바이스의 어플리케이션은 용량과 기능성을 제한했다. 그러나 하드웨어가 발전함에 따라 더 높은 대역폭을 사용하여 통신이 가능해지면서 모바일 디바이스 어플리케이션은 더욱 복잡해졌다. 또한, 다른 디바이스와 다른 서버 기반 어플리케이션 사이의 복잡한 통신에 대한 더 많은 수요가 발생한다. 이러한 경우에 복잡한 machine-to-machine 상호작용은 모바일 어플리케이션 통합 요구를 필요로 한다[5].

모바일통합 아키텍처는 무선 포탈 네트워크(Wireless Portal Network). 확장된 무선 인터넷(Wireless Extended Internet) 그리고 P2P(Peer-to-Peer) 3가지 유형을 나타낸다[6, 7].

무선 포탈 네트워크는 WAP 게이트웨이를 이용하고, 서비스 요구는 WML 메시지에 포함되고 포털/게이트웨이가 메시지를 수신하면 SOAP 요구로 번역한다. 웹 서비스의 응답은 WML으로 결과를 번역하고 모바일 디바이스에게 결과를 돌려준다. 확장된 무선 인터넷은 본 논문에서 제안한 시스템과 같은 형태로 모바일 디바이스가 실제 SOAP 웹 서비스 클라이언트 역할을 담당하며 모든 통신은 XML과 HTTP를 이용하고 TCP/IP 연결은 모바일 디바이스로부터 셋업된다. P2P에서 모바일 디바이스는 웹 서비스 호스트이며 다른 디바이스와 상호 작용한다.

2.3 웹 서비스 보안

서버와 클라이언트의 보안을 위해서 사용자 인증(Authentication), 권한 설정(authorization), 감사(auditing)와 로그(logging), 저장 정보의 무결성, 디지털 서명(digital signature), 메시지 암호화(encryption) 등이 이용된다[8].

웹 서비스 보안은 트랜스포트 레벨 보안과 애플리케이션 레벨 보안으로 분류한다. 트랜스포트 레벨 보안은 웹 서비스 프로토콜의 트랜스포트 계층 자체의 보안으로 SOAP 메시지는 트랜스포트 계층의 패키징 메커니즘을 이용해서 SOAP 메시지에 직접적인 영향을 주지 않으면서 캡슐화된다. 트랜스포트 수준의 보안은 point-to-point 보안이므로 웹 서비스와 같은 SOAP 중개자가 존재하는 경우는 XML 기반의 보안 기술을 포함하는 애플리케이션 수준의 보안이 필요하다.

트랜스포트 레벨 보안의 SSL(Security Socket Layer)는 point-to-point 인증과 데이터의 기밀성 그리고 메시지의 무결성을 보장하지만 전송 부분만을 담당하고 단순한 연결 보안으로 접속 권한 정보 지원이 불가능하므로 SSL 채널을 벗어난 메시지는 암호화 과정을 생략하므로 여러 단계를 경유하는 트랜젝션의 경우 안전 보장이 불가능하다[9].

또한 다른 형태의 트랜스포트 레벨 보안은 end-to-end 보안으로 SOAP 중개자가 존재하고 트랜스포트 수준의 보안 외에 XML 기반의 보안 기술을 포함하는 애플리케이션 수준의 보안이 필요하다. 트랜스포트 수준의 보안을 구현한 대표적인 것이 IPSec(Internet Protocol Security)이다. IPSec은 암호화 기법과 IP 인증, 프라이버시, 데이터 무결성을 구현한다[2, 10].

애플리케이션 레벨의 보안은 XML 메시지 보안으로 XML 서명과 XML 메시지 암호화 등이 존재한다. XML 서명은 XML 트랜잭션에서 이용 가능하게 설계된 디지털 서명으로 디지털 서명 오퍼레이션의 결과에 대한 스키마를 정의하고, 인증과 데이터 무결성, 서명 데이터의 거절 거부 등을 지원한다[11].

XML 메시지 암호화와 XML 서명의 기본적인 특징은 XML 문서 전체 또는 일부에 대한 서명이 가능하다. XML 기반의 SOAP 메시지는 어떠한 프로토콜을 통해서도 전송이 가능하다. 모든 웹 서비스 오퍼레이션에 사용자 이름과 암호를 매개변수로 추가하여 전송하면 웹 서버에서 수신된 매개

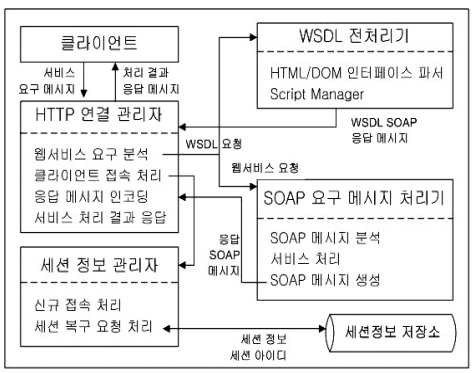
변수에 의해 사용자를 확인하고 실행한 결과값을 SOAP 응답 메시지로 리턴한다.

3. 모바일 웹 서비스 시스템 설계 및 구현

본 논문에서는 모바일 이동 통신에서 발생하는 순간적인 단절의 정도를 분석하기 위하여 표준 통신 프로토콜을 적용한 모바일 웹 서비스 시스템을 구축하고 모바일 환경을 위한 모바일 웹 서버로 확장하여, 모바일 환경에서 SSL, S-HTTP를 적용하여 발생하는 보안 단절에 대한 분석을 진행하고 보안 안전성을 위한 모바일 연결성 관리 기술을 적용한다.

3.1 시스템 구성

본 논문에서는 모바일 환경에서 발생하는 단절 상황에서 보안성 향상을 위해 인증된 보안 세션 관리를 담당하는 XML/SOAP 메시지 처리 기반의 보안 세션 관리 프레임워크를 제안한다. 제안 시스템 구성은 다음 (그림 1)과 같다.



(그림 1) 제안 시스템의 구성 요소

본 논문에서 제안한 시스템은 모바일 환경에 기존의 HTTP 기반의 웹 콘텐츠 요구에 대한 웹 서비스를 제공하기 위하여 SOAP 서비스를 지원하고 시스템의 확장성과 유지 보수를 위하여 Apache 그룹의 AXIS를 기반으로 설계한다. 따라서 본 논문에서는 톰캣(Tomcat) 서블릿 엔진을 사용하지 않고 웹 서버 접속 시간 단축과 지연 처리 향상을 위하여 직접 SOAP 요구와 응답 메시지를 처리하는 시스템을 설계하고 구현한다.

3.2 HTTP 연결 관리자

HTTP 연결 관리자는 SOAP 메시지를 분석하는 모듈과 콘텐츠 웹 서버로부터 수신 결과를 규격화된 SOAP 메시지로 구현하는 모듈이 필요하다. 따라서 웹 서비스 요청을 수신하는 SOAP 메시지 분석기 그리고 SOAP 메시지 생성기로 구성하는 SOAP 메시지 요구 처리기를 구현한다. 클라이언트의 요청 형태가 WSDL 요청인 경우는 WSDL 전처리기에 명령을 전달하고, 웹 서비스 요청인 경우는 SOAP 메시지 요구 처리기에 메시지를 전달한다. 그리고 처리 결과는 인코딩하여 클라이언트로 다시 전달한다.

3.3 WSDL 전처리기

WSDL 전처리기는 기존 웹에 존재하는 HTML 문서를 무선 통신 데이터 형식인 WSDL 메시지 형식으로 변환하기 위하여 WSDL 전처리기를 설계한다. 즉, 클라이언트의 웹 서비스 요구에 대한 특정 스크립트를 WSDL 파일로 생성한다. 다음의 (그림 2)는 클라이언트의 요청 서비스 에 대한 수행 과정을 나타낸다.

클라이언트의 전송 정보에 세션 정보가 존재하는 경우는 세션을 복구하고, 존재하지 않으면 세션을 생성하여 서비스 실행 결과를 SOAP 메시지 생성부로 전달한다. 클라이언트가 HTTP를 통해 웹서버에 접속하고 클라이언트의 요청 서비스에 대한 XML SOAP 메시지를 전달하고, 요청이 웹

```
Procedure Web_Service_Request_Transaction()
{
  read WSDL name of Client request service
  creation of SOAP information from XML SOAP message
                                          of Client
  an extract BODY from SOAP information
  an extract service name of BODY;
  an extract name and value of MessageElement in BODY
  creation of input binding by input name and value
  if(name of input value = in_test(session_id)) then
            setting up to session ID
  if(session ID = setting)) then session recovery
  else {
          creation of session number
          creation of session information
          creation of session ID
          storing in session registry
        }
  execution of client request service ;
}

Procedure Execution_Result _Information_Creation()
{
  creation of response header information
  creation of response SOAP message information
  if(result = normal)) then
  {
    creation of body information
    creation of result value information
    creation of session ID information
    adding created informationto SOAP message
   }
   else
       creation and adding of a failure message information
  creation of result message
  transmission XML SOAPmessage to Client
  connection termination
}
```

(그림 2) 클라이언트의 요청 서비스 수행 과정

서비스 요청으로 확인되면, 클라이언트가 요청한 서비스가 포함된 WSDL명을 가져온다. 그리고 클라이언트의 XML SOAP 메시지로부터 SOAP 정보를 생성하고, 생성한 SOAP 정보로부터 BODY 부분에서 클라이언트가 원하는 서비스명에서 입력값으로 입력한 MessageElement에서 입력값의 이름과 값을 가져온다. 입력값의 이름과 값으로 입력 바인딩을 생성하고, 입력값의 이름이 in_test (session_id)이면 세션아이디로 설정한다.

그리고 세션아이디가 이미 설정되어 있으면 세션을 복구하고, 설정되어 있지 않으면 세션 번호, 세션 정보, 세션 아이디를 생성하고 세션을 관리하기 위하여 저장한 다음 클라이언트가 요청한 서비스를 수행한다.

본 논문에서 제안한 WSDL 전처리기는 HTML /XML 파서를 구현하여 WSDL 생성을 지원하고, HTML/XML 파서를 통하여 클라이언트로부터 요청된 메시지를 WSDL로 간단하게 변환하며, Java 1.4의 org.w3c.dom 라이브러리를 이용하여 AXIS에 독립적으로 실행하며,Dom의 Document 객체를 사용하고, 모든 태그는 Element 객체를 사용한다.

WSDL 전처리기는 클라이언트의 요청 메시지에서 WSDL명을 추출하고, 응답 헤더 정보를 생성하고, 추출한 WSDL명에 대한 WSDL 정보를 획득한다. SOAP 메시지로 사용할 XML 문서 정보를 생성하고, 추출한 WSDL의 정보를 WSDL에 추가하고, 출력형에 대한 complexType도 추가한다. 헤더 정보를 추가하여 처리 결과를 생성하고 연결 관리자로 전달하고 클라이언트가 결과를 수신하면 WSDL 요청을 완료한다.

3.4 SOAP 메시지 요구 처리기

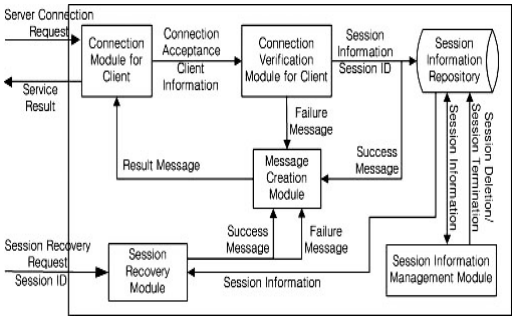
SOAP 메시지 분석기는 HTTP를 통하여 요청 서비스명, 입력값, 세션 아이디가 포함된 클라이언트의 SOAP 메시지 요청을 접수하고 분석하여, 요청한 서비스의 WSDL명을 추출한 후 분석한 정보에서 WSDL명, 서비스명, 입력 값들을 추출한 후 서비스 처리부에 서비스 처리를 요청한다.

SOAP 메시지 요구 처리기는 클라이언트가 요청한 서비스에 대한 정보를 획득하고 서비스를 처리한다.

SOAP 메시지 생성부에서는 서비스 처리부에서 처리한 서비스 결과를 WSDL에 일치한 응답 SOAP 메시지 생성을 위하여 응답 헤더 정보와 응답 SOAP 메시지에 대한 정보를 생성하고, 서비스 처리 결과가 정상인지 확인하여, 정상이면 서비스 결과값과 세션 아이디를 추가하고, 정상이 아니면 실패 정보를 추가하고, SOAP 메시지 정보로 SOAP 메시지를 생성하여 접속부로 전송하면, 접속부는 다시 클라이언트로 전달한다.

3.5 세션 정보 관리자

세션 정보 관리자는 서버와 클라이언트간의 접속 상태 유지를 위해 필요한 모든 정보를 저장하고 관리한다. 클라이언트는 TCP/IP 프로토콜을 사용하여 서버에 접속하고, 세션 정보 관리자는 클라이언트가 접속하면 세션 정보를 생성하여 세션 정보 저장소에 저장하고, 클라이언트의 세션 복구 요청이 있으면 클라이언트의 세션 복구 요청을 처리하고, 사용하지 않는 세션 정보의 관리와 처리를 담당한다. 다음의 (그림 3)은 세션 정보의 관리 및 처리를 나타낸다.



(그림 3) 세션 정보의 관리 및 처리 절차

세션 정보 관리자는 접속한 클라이언트의 서버 이용 가능 여부를 현재의 클라이언트 수와 최대 수용 가능한 접속수를 비교하여 판단한다.

클라이언트가 서버의 이용이 가능하면, 세션 관리 에이전트에서는 접속한 클라이언트에 대한 정보를 확인하고, 확인된 정보를 사용하여 세션 정보를 생성하고 클라이언트에 부여할 세션 아이디를 생성하면, 생성된 세션 정보를 생성된 세션 아이디에 매핑시켜 세션 정보 저장소에 저장하고, 세션 아이디를 포함한 접속 성공 메시지를 생성하지만 클라이언트의 서버 이용이 불가능하면 접속 실패 메시지를 생성한다.

생성된 메시지는 클라이언트로 전송하고, 클라이언트는 결과 메시지를 수신하여 성공 메시지인지 판단하여 성공 메시지인 경우는 차후 세션 복구가 필요할 때 사용할 세션 아이디를 기억하고 실패 메시지인 경우는 서버와의 연결을 종료한다.

만약에 의도하지 않은 상황에 의해서 서버와 클라이언트의 연결이 끊어진 경우에는 클라이언트가 이전의 세션을 복구한다.

4. 실험 및 평가

본 논문은 안전한 모바일 웹 서비스 제공을 위해서 상호 운영성 기반의 확장성, 신뢰성과 보안성 등을 고려한다. 또한 무선 환경의 제약 사항을 고려하면서 전송 계층 및 응용 계층에서 안전성을 제공한다.

4.1 실험 환경

본 논문의 실험은 4가지의 웹 서비스 시스템을 각각 구축하고 제안 시스템의 웹 서비스와 비교/평가한다. 실험에 사용된 시스템은 AXIS를 기반으로 다음과 같이 구성한다.

- 표준 System : AXIS + 톰캣 5.5 사용 (기존의 시스템)
- Test 1 System : AXIS + WSDL 전처리기
- Test 2 System : AXIS + 세션 정보 관리자
- 제안 System : AXIS + WSDL 전처리기 + 세션정보관리자 사용

실험을 위해서 SOAP 요구 메시지 생성기는 여러 개의 문자 전송과 다수의 동시 요청을 수행하는 역할을 담당하며, 1GB RAM, 인텔 펜티엄4 2.8GHz의 클라이언트 4대를 사용한다.

본 논문의 제안 시스템의 성능 평가를 위해 AXIS를 기반으로 4가지의 웹 서비스 실험 시스템을 구성하였다. 각각의 실험 시스템을 위해 2GB RAM, 4-way Intel Xeon 3.0GHz Windows 2000

서버를 사용한다.

실험을 위한 가정으로 SOAP 요구는 4대의 모바일 클라이언트 시뮬레이션 프로그램에 의해 실험 시스템에 연결하고, 1초에서 10초의 시간 간격으로 여러 개의 SOAP 요구 메시지를 반복 요청한다. 연결이 종료되면 시뮬레이션 프로그램은 실험 시스템에 연결을 위해 반복적인 연결 시도를 한다.

시스템에 접근하는 사용자는 동시에 200명으로 가정하고, SOAP 요구는 4대의 클라이언트에 의해 50개의 쓰레드를 생성시켰다. 본 논문의 실험에 이용된 콘텐트 웹 서버는 범용의 웹 사이트를 선택하였다. 서버 타임아웃은 각각 30초이다.

4.2 평 가

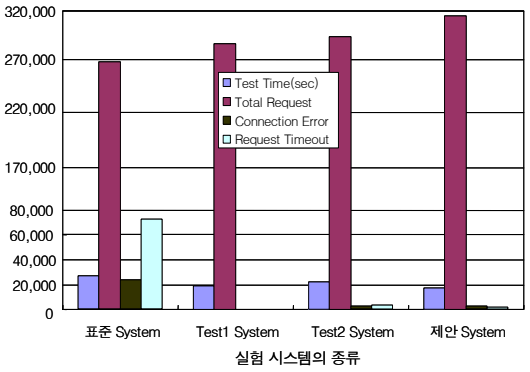
본 논문의 실험 과정은 클라이언트가 SOAP 형식으로 웹 서비스를 요청하는 경우, 실험 시스템은 SOAP 메시지와 내용을 HTTP 호출로 웹 서버에 전달한다. 콘텐츠 웹 서버가 전송한 HTTP 응답을 실험 시스템이 수신하면, WSDL을 생성시키고 클라이언트인 모바일 장치에게 SOAP 응답 메시지를 전송한다.

〈표 1〉 동시 요청의 실험 결과

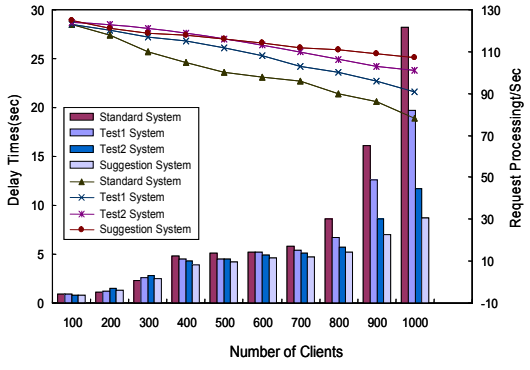
	표준 System	Test 1 System	Test 2 System	제안 System
Test Time(sec)	27,720	19,281	24,053	17,640
Total Request	267,253	285,159	292,108	314,744
Connection Error	25,465	7,699	8,913	2,146
Request Timeout	71,935	34,704	3,418	1,454

<표 1>은 실험 결과를 나타낸다. 본 논문의 실험을 위한 4개의 구현 시스템 비교를 위하여 실험 시간 동안에 발생한 클라이언트의 전체 요구의 개수, 타임아웃 기간에 연결을 실패한 요구의 개수, 서버의 busy 상태로 인한 연결이 실패한 개수, 세션 연결 오류 개수, 서버 연결을 실패한 클라이언트 개수 그리고 클라이언트의 요청에 대하여 타임아웃이 초과된 횟수와 같은 정보를 수집한다. 본 논문은 모바일 웹 서비스 시스템의 실행 평가를 중심으로 실험을 진행하였다.

제안된 시스템은 모든 항목에서 성능 향상이 이루어졌다. 비록 제안된 시스템의 test time이 표준 시스템의 test time보다 짧았지만 전체 요청 횟수는 표준 시스템보다 많았고 타임아웃 개수는 표준 시스템보다 적었다. 제안 시스템에서의 초당 평균 요구 수는 17.84이지만 표준 시스템에서는 9.64이다. 표준 시스템에서는 많은 연결 에러가 발생한다.



(그림 4) 각각의 시스템별 오류 발생 횟수



(그림 5) 클라이언트 수와 지연 시간과 처리량

(그림 4)와 (그림 5)의 실험 결과는 서블릿 컨테이너를 사용하는 것보다 본 논문에서 제안한 시스템의 효율이 높다는 것을 의미한다.

그리고 웹 서비스의 성능 향상은 전체 시스템의 성능 향상을 가져와 누적되는 연결 신호 처리의 향상을 의미한다. 결국 연결 요청 횟수가 많아질수록 기존의 웹 서비스 시스템의 오류 횟수는 기하급수적으로 증가한다.

5. 결 론

웹 서비스 기술의 발전으로 다양한 기술 적용이 가능하고, 보안 요구 사항도 증가하고 있다. 특히, 현재 많은 비즈니스 어플리케이션들은 웹 서비스를 통한 경쟁력 있는 서비스와 안전성이 요구되고 있으며, 모든 트랜잭션에서 프라이버시에 대한 보안의 중요성 역시 점차 증가하고 있다.

본 연구에서는 이러한 시대적 요구 사항들에 부응하기 위해 강력하고 확장성, 유연성을 갖춘 보다 안전한 웹 서비스 보안 기술과 정보 보호 문제의 해결을 제시하였다.

참 고 문 헌

[1] F. Ishikawa, N. Yoshioka, Y. Tahara, and S. Hondien, “Mobile Agent System for Web Services Integration in Pervasive Networks”, (IWUC 2004), pp. 38-47, April, 2004.

[2] 정지훈, 웹 서비스, 제 23장, 한빛미디어, 2002.

[3] Douglas B. Terry, Venugopalan Ramasubramanian, Caching XML Web services for mobility, ACM Queue－Tomorrow’s Computing Today, Vol. 1, No. 3, pp. 70-78, 2003.

[4] Guido Gehlen and Ralf Bergs, Performance of mobile web service access using the wireless application protocol(wap), In Proceedings of World Wireless Congress 2004, pp. 427-432, San Francisco, USA, 05 2004. University Aachen, Communication Networks, 2., 3.1.

[5] Robert Steele, “A Web Services-based System for Ad-hoc Mobile Application Integration”, Proc. Of IEEE International Conference on Information Technology : Computers and Communications, pp. 248-252, 2003.

[6] M. Juntao Yuan and J. Long, “Java Readies itself for wireless Web Services”, JavaWorld Magazine Article, June 2002, http://www.javaworld.com/javaworld/jw-06-2002/jw-0621-wireless.html.

[7] Sangyoon Oh, WEB SERVICE ARCHITECTURE FOR MOBILE COMPUTING, Department of Computer Science, Indiana University August 2006.

[8] 박배효, 이재일, “안전한 웹 서비스를 위한 SOAP 메시지 보안기술 연구”, 한국정보보호학회지, 제14권, 제4호, pp. 10-18, 2004.

[9] V. Beltran, D. Carrera, J. Guitart, J. Torres, and E. Ayguad’e, “A Hybrid Web Server Architecture for Secure e-Business Web Applications”, LECTURE NOTES IN COMPUTER SCIENCE(LNCS), Springer Berlin /Heidelberg, pp. 366-377, 2005.

[10] 김배현, 유인태, “웹 서비스 보안 기술”, 인터넷정보학회지, 제6권, 제2호, pp. 16-23, 2005.

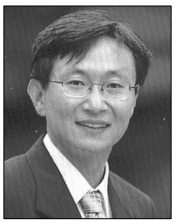
[11] 최동희, 박석, “접근제어 정책구현을 위한 역할 기반 XML 암호화”, 정보보호학회 논문지, 제15권, 제1호, pp. 3-15, 2005.

김 용 태

1988년 숭실대학교 전자계산학과 (공학석사)
2008년 충북대학교 전자계산학과 (이학박사)
2002년~2005년 (주)가림정보기술 이사
2006년~현재 한남대학교 공과대학 멀티미디어학부 강의전담교수
관심분야 : 모바일 웹 서비스, 정보보안, 센서 웹, 모바일 통신보안, 멀티미디어

정 윤 수

2000년 충북대학교 전자계산학과 (공학석사)
2008년 충북대학교 대학원 전자계산학(이학박사)
관심분야 : 센서 보안, 암호이론, Network Security, 이동통신 보안



박 길 철

1986년 숭실대학교 전자계산학과 (석사)
1998년 성균관대학교 전자계산학과(박사)
2006년 UTAS, Australia 교환교수
1998년~현재 한남대학교 멀티미디어학부 교수
관심분야 : Mobile Communication and Multimedia, Network Security