SSL을 이용한 안전한 전자상거래 설계 및 구현
-A Design and Implementation of Secure Electronic Commerce Using SSL-

유현우*, 임경묵*, 곽승욱*, 하재승*, 박경배**, 이광배*, 김현욱*
Hyun-Woo Yoo*, Kyoung-mook Yim*, Seung-Uk Kwak, Jai-Seung Ha*,
Kyoung-Bae Park**, Kwang-Bae Lee*, Hyen-Uk Kim*

요 약

월드 와이드 웹(WWW)의 발달로 인터넷을 기반으로하는 전자상거래(Electronic Commerce)이용등 전세계적으로 네트워크의 사용이 늘어나게 되었고, 더불어 네트워크에 대한 보안과 암호화의 문제가 이슈로 떠오르고 있다. 하지만 신뢰감 부족으로 인하여 인터넷을 통한 제품 구매를 선호하지 않는다. 따라서 본 논문에서는 이러한 문제점을 개선하기 위해 웹서버와 클라이언트사이에 SSL(Secure Socket Layer)을 사용하는 전자상거래 시스템을 구현하여 보았다. 이것은 전자화폐의 지불과정이나, 거래자의 인증과정에 앞서서 수행되므로, 아직 보안상 불완전한 거래에 사용자들로 하여금 신뢰를 줄 수 있다.

제 1 장 서 론

1970년 ARPANET(Advanced Research Projects Agency Network)은 국방용으로 사용되기 위해 시작되었고, 소수의 연구소, 대학, 공공기관에서만 사용되었다. 그후 TCP/IP 기반하의 인터넷으로 개방되면서 가정과 대학, 기업, 연구소, 그리고 정부 연구기관등 서로 연결되어있는 전 세계적 규모의 분산 시스템으로 발전하였고, 이로 인해 많은 사용자들이 네트워크를 통해 서로 정보를 공유하고 사용할 수 있게 되었다. 이러한 웹의 발달로 인터넷을 통한 정보의 전달이나, 전자상거래(Electronic Commerce)등 네트워크의 사용이 늘어나게 되었고, 더불어 네트워크에 대한 보안과 암호화가 최근의 이슈로 떠오르고 있다.
하지만, 현재 쇼핑몰의 웹사이트가 취약하며, 소비자를 유치하기 위한 다양한 기능 제공과 안정성이 부족한 상태이다. 안전한 전자상거래를 구현하기 위해서는 웹의 보안이

* 명지대학교 전자공학과 멀티미디어 연구실 · ** 여주대학 전산정보처리학과

강화되어야 한다.
이에 본 논문에서는 전자상거래 시스템 해킹으로 인한 상거래 정보의 위 · 변조, 시스템 마비 및 개인정보 유출 등을 방지하기 위하여 웹서버와 클라이언트사이에 안전한 통신채널을 제공하는 SSL을 사용하여 시스템을 구현하고자 한다. 이로써 사용자에겐 안정성과 정당성을 부여하고 서비스 업체나 서버 관리자에겐 불법 접속으로 인한 제3자의 데이터이용과 악용등을 최소화하는 시스템을 제안한다.
본 논문의 구성은 다음과 같다. 2장에서는 관련연구로서 전자상거래의 시스템 구성과 문제점을 살펴보고, 3장에서는 웹의 보안위협요소와 SSL의 기본원리를 설명한다. 4장에서 SSL을 사용한 시스템 구축하여 시뮬레이션과 분석을 하고, 그리고 5장에서 결론을 맺는다.

제 2 장 전자상거래 시스템

2.1 전자상거래 개요

Electronic Commerce(EC, 전자상거래)라는 용어는 1989년 미국의 Lawrence Livermore National Laboratory에서 사용된 후 1993년에 미국 연방정부가 도입하면서 확산된 것으로 추정된다.[8] 미국에서 본격적으로 인터넷 상점이 시작된 시기는 Netscape Browser가 도입되면서 인터넷이 비즈니스용도로 쓰이기 시작한 1994년 말부터 1995년 초가 된다[9]
전자상거래는 일반적으로 기업, 정부기관과 같은 독립된 조직간 혹은 조직과 개인간에 다양한 전자적인 매체를 이용하여 상품이나 서비스를 교환하는 방식이라고 정의할 수 있다.

2.1.1 전자상거래의 거래 유형

전자상거래의 거래 유형은 다음의 4가지로 구분 할 수 있다.

① 기업과 기업 (Business to Business)
② 기업과 개인 (Business to Consumer)
③ 기업과 정부 (Business to Government)
④ 기업내 거래 (Business in Enterprise)

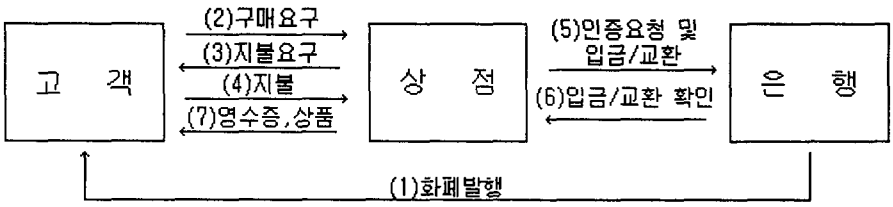


그림 1 전자상거래의 기본 절차

2.2 전자지불

발달로 인터넷상에서 상품의 대금결제 수단으로써 전자화폐를 사용한 전자지불방법이 등장하고 있다. 전자지불단계는 전자상거래에서 중요한 단계중 한가지인데, 이러한 전자지불의 기본구성은 고객, 상점, 은행사이에 발생하는 발행(Withdrawal), 지불(Payment), 예치(Deposit)로 이루어진다.

전자지불의 기본적인 구성도는 다음과 같다

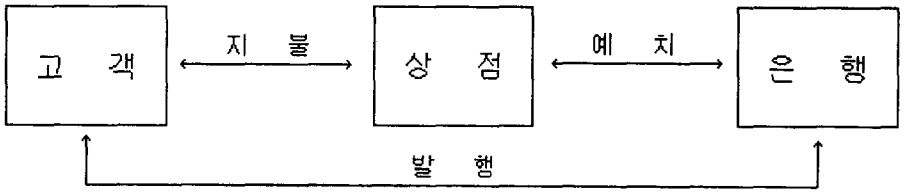


그림 2 전자지불의 구성도

전자지불의 방법은 특성에 따라 표 1과 같이 분류할 수 있다.

표 1 전자지불 시스템

형 태	종 류
전자화폐	E-Cash, Net cash
신용카드기반	First Virtual, Cybercash, iKP, SET
전자수표	FSTC의 E-check, NetCheque

2.2.1 전자화폐

전자화폐는 인터넷상에서 현금과 동일한 가치를 가지고 있어야 하며, 주로 소액거래에 사용된다. 그러나 매 사용시 고객의 정보가 저장된다면 개인의 프라이버시가 침해될 수 있으므로 익명성을 보장해야 한다.

전자화폐 시스템에서 제공해야 하는 요구사항을 다음과 같이 요약할 수 있다.
• 안정성, 익명성, 이중사용, 양도성, 휴대성, 분할이용

2.2.2 신용카드 기반

신용카드를 사용하는 지불 방식은 실세계의 지불 과정과 유사한 절차로 이루어지므로 트랜잭션 비용보다 거래금액이 크다. 하지만 전자상거래에서 이루어지는 거래는 소액거래가 많이 있으므로 트랜잭션 비용을 줄이는 것이 필수적이다.

2.2.3 전자수표

전자상거래에서는 거래하려는 상대방의 얼굴과 신원을 확인 할 수 없으므로, 사용자의 인증과정이 반드시 필요하다. 또한 트랜잭션 비용이 높지만 큰 액수의 사용에 적합하다. 종이수표와 비교할 때 비용이 적으므로 소액거래에도 사용이 가능하다. 단점은 수표의 중복발행 및 인증을 위해 수표의 사용에 관한 유효일자가 정해져 있다는 것이다.

2.2.4 국내의 전자지불

● 이니페이
보안전문 업체인 이니시스에서는 RSA암호화 기법을 이용해 1024bit의 키 길이를 가지며 신용카드, 직불카드, 계좌이체 등 다양한 지불방식을 지원하는 국내 금융환경에 적합한 전자지불 S/W를 개발하여 서비스중에 있다. [3][4]

● 아이캐시
1998년말 동성정보통신이 네트워크형 전자화폐 아이캐시를 선보였다. 아이캐시는 조흥은행, 데이콤, IBM, 동성정보통신, 오라클, 대홍기획, LG정보통신 등 7개사가 인터넷 EC에서 수수료없이 소액결제를 하기 위해 추진해온 IC카드 기반 전자화폐로 개인키, 계좌번호, 인증서, 패스워드 등에 대한 비밀정보를 암호키를 사용해 보안기능을 대폭 강화한 것은 물론 사용자 인증과 전자서명 등의 EC를 위한 핵심 기능도 제공한다. [2][4]

제 3 장 웹 보안 위협 분석

3.1 웹 보안 개요

웹은 기본적으로 클라이언트-서버 모델을 기본으로 하고 있다. 인터넷 이용자들은 보통 브라우저 또는 클라이언트라고 하는 응용(application)을 웹 서버에 연결함으로써 사

용이 가능하게 된다. [1]
인터넷은 원래 개방성에 근간을 두고 있으므로, 보안상의 문제는 어느 누구도 피할 수 없다. 웹은 이러한 인터넷에 기반을 두고 있으므로 보안의 측면에서 매우 취약하다. 이 장에서는 웹에서 발생할 수 있는 보안 위협과 현재 대응방법에 대해 알아보고, 취약성을 보완하기 위해 웹 상에서 적용되고 있는 프로토콜에 대해 소개한다. [7]

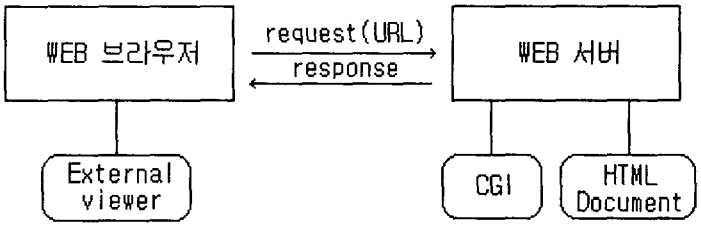


그림 3 웹의 개념

3.1.1 웹 보안 위협과 대응책
웹 상에서 전송되는 메시지에 대한 제 3자의 위협을 보면 수동적 공격(passive attack)과 능동적 공격(active attack)이 있다.
● 수동적 공격은 단지 도청(eavesdropping)이나, 트래픽 분석(traffic analysis) 등 메시지의 변형 없이 가해지는 공격이다.
● 능동적 공격은 변조(modification), 삽입(insertion), 삭제(delay), 재생(replay) 공격 등 메시지의 변형과 조작을 가하는 공격이다.

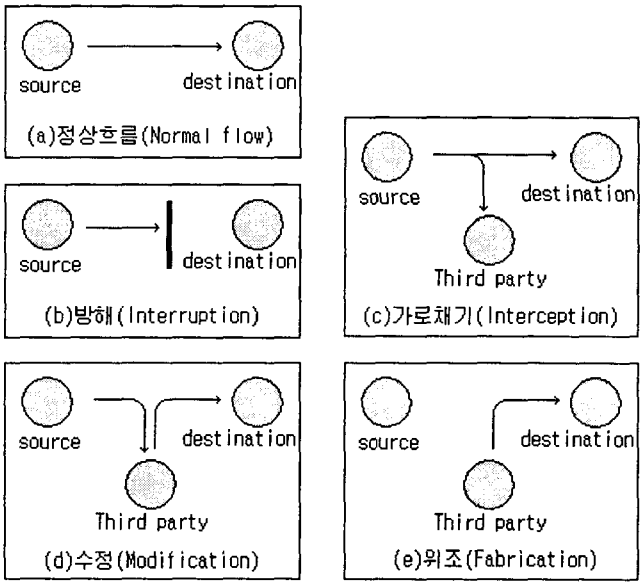


그림 4 보안 위협

위와 같은 공격에 대응하여 제공되는 보안 서비스는 다음과 같다.

1) 무결성(Integrity)
무결성은 메시지의 내용을 생성, 변경, 삭제 할 수 없도록 보호하는 기능을 한다.

2) 기밀성(Confidentiality)
전송되는 메시지가 외부에 노출되어 단지 클라이언트와 서버 두 통신 당사자만이 알아야 하는 메시지를 제 3자가 알 수 있다.

3) 부인봉쇄(nonrepudiation)
서버와 클라이언트사이에 발송된 메시지를 부인하거나, 전송하지 않은 메시지를 받았다고 주장할 수 있고, 또는 수신사실을 부인할 수 있다.

4) 인증(Authentication)
서버와 클라이언트가 서로 통신을 하려고 할 때, 각각에 대해서 실제 통신하려던 정당한 서버와 클라이언트인지 확인해야 한다.

5) 접근제어(Access Control)
클라이언트에서 통신채널을 통한 서버자원의 무분별한 사용을 금지하기 위해 서버의 시스템이나 응용프로그램의 접근을 통제하거나 사용범위를 제한한다.

6) 가용성(Availability)
가용성은 네트워크에 접속된 전체 시스템의 성능을 안정적으로 유지하는 한편 전체시스템의 시용 효율에는 이상이 없도록 하는 것이다.

3.2 웹 보안 프로토콜

현재 웹을 이용하여 통신하는 서버와 클라이언트사이에서 각 레벨에 적용되는 보안 프로토콜들은 다음과 같다.[7]

- Network Level : IPSec,
- Transport Level : SSL 또는 TLS,
- Application Level : Kerberos, S/MIME, PGP, SET, S-HTTP등이 있다.

다음 절 에서는 본 논문에서 구현에 적용 할 Transport Layer에서의 대표적 보안 프로토콜인 SSL에 대해 알아본다.

3.2.1 SSL(Secure Socket Layer)

SSL은 Netscape 사에서 인터넷 보안을 위하여 개발한 프로토콜이다. RSA와 X.509를 사용하여 데이터의 암호화, 서버 인증 기능, 데이터의 무결성(integrity) 그리고 선택적으로 클라이언트 인증 기능을 수행하며 응용 계층과 TCP/IP 계층의 사이에 위치한다. 응용 계층의 아래에 위치하기 때문에 각종 응용 계층 프로토콜(HTTP, TELNET, FTP 등)에서 사용이 가능하다.
그림5와 같이 SSL은 TCP/IP 또는 그 밖의 Transport Layer위에 하나의 layer를 따로 구현한 것으로, 통신하고자 하는 서버와 클라이언트사이에 소켓 루틴들을 사용하여 먼저 안전한 통신 채널을 확립한 다음, 이 채널을 통해 암호화된 정보를 교환할 수 있도록 하고있다. SSL의 경우, TCP/IP 연결을 시작했을 때, 보안 응답 확인방식(handshake)을 사용하여 보안 등급을 협상한다. SSL을 사용하려면 URL에 HTTPS라는 액세스 방식을 사용해야 하는데 이 경우 port 443을 사용한다. 보통 HTTP에서는 port 80을 사용하므로 이렇게 port를 구분하여 사용하면 하나의 서버 호스트에서 보안 웹 서버와 보안 기능이 없는 서버를 동시에 실행할 수 있다.
SSL프로토콜의 구성을 보면 SSL은 그 안에서 SSL Record Protocol, SSL Handshake Protocol, Change Cipher Spec Protocol, Alert Protocol로 나누어진다. SSL Record Layer에서는 암호화 처리를 위한 기본 단위들로 나누고 그것을 구분하는 것을 담당하고, SSL Handshake Layer는 암호화 방법이나 열쇠의 결정 및 협상을 담당한다.[5][6][7]

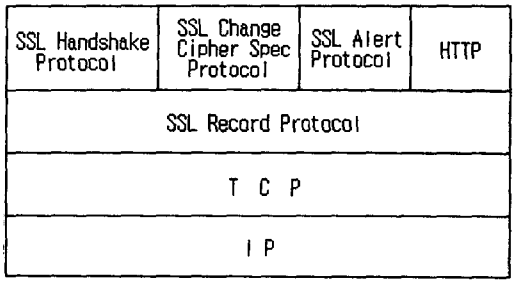


그림 5 SSL 프로토콜

3.2.2 SSL Record Protocol

레코드 프로토콜은 상위 계층의 프로토콜들을 캡슐화하기 위해 사용되는데 이것은 TCP 상위에 존재한다. DES(Data Encryption Standard), RC4(Rivest Cipher version four, Ron's Code) 등과 같은 대칭키 암호화 알고리즘을 사용하여 두 통신 개체간의 비밀보장 서비스가 제공되며, 접속 과정에서 클라이언트와 서버 인증이 제공된다.

3.2.3 SSL Handshake Protocol

SSL이 서버와 클라이언트 사이에서 동작하기 위해서 필요한 사전작업 즉, 상호간 인증, 암호화방법, MAC 알고리즘, 암호키결정 등의 작업들이 handshake protocol에서 수행된다. handshake protocol은 클라이언트와 서버에 의해 교환되는 메시지로 구성되는데 각 메시지는 그림7(c)의 세 개의 필드를 포함한다.

- Type(1 byte) : 표 1에 정의된 10개의 메시지중 하나
- Length(3 byte) : 메시지의 길이
- Content(≥1 byte) : 메시지와 관련된 파라미터

다음은 handshak protocol에서 수행되는 작업들을 4개의 단계별로 나눈 것이다.

■ 1단계 - Establish Security Capabilities

► Logical Connection을 초기화하고, Security Capabilities를 설정

► Client hello

· 최상위 SSL version, 32-bit의 timestamp와 28byte의 random number로 구성된 32byte initial random number, Session ID, CipherSuite, Compression Method등 5개의 파라미터가 포함된 client hello 메시지를 보내는 클라이언트에 의해서 교환이 초기화된다.

■ 2단계 - Server Authentication and Key Exchange

서버가 인증을 위해 자체의 certificate를 보냄으로 2번째 단계가 시작된다.

► Certificate 메시지

· 하나 또는 여러개의 X.509 certificate로 구성.

· Fixed DH의 키 교환 용도로 사용.

► Server key exchange 메시지

· 자체 certificate가 없거나, 단지 서명에만 사용되면 server key exchange 메시지가 보내지게 된다.

· fixed DH 파라미터를 가진 certificate를 보낼 때 이 메시지는 사용되지 않는다.

► Certificate request 메시지

· non-anonymous 서버는 클라이언트로부터 certificate를 요청.

· certificate type, certificate authorities등 2개의 파라미터를 포함.

► Server done 메시지

· 파라미터를 가지지 않는다.

· server hello의 끝을 알린다.

· 이 메시지 후에 클라이언트의 응답을 기다린다.

■ 3단계 - Client Authentication and Key Exchange

- Certificate 메시지
  - 서버의 certificate 요청이 있는 경우 보내진다.
  - 적절한 certificate가 없으면 경고 메시지, no certificate를 대신 보낸다.
- Client key exchange 메시지
  - server key exchange에서 선택된 공개키 알고리즘에 의해 메시지가 결정된다.
- Certificate verify 메시지
  - 클라이언트 certificate의 확실한 증명을 제공하기 위해 보내진다.

■ 4단계 Finish

secure connection의 setting을 완결하는 단계

- Change cipher spec 메시지
  - 클라이언트가 change cipher spec 메시지를 보내고, pending CipherSpec을 current CipherSpec에 복사한다.
- Finished 메시지
  - key exchange와 인증이 완료되었음을 의미.

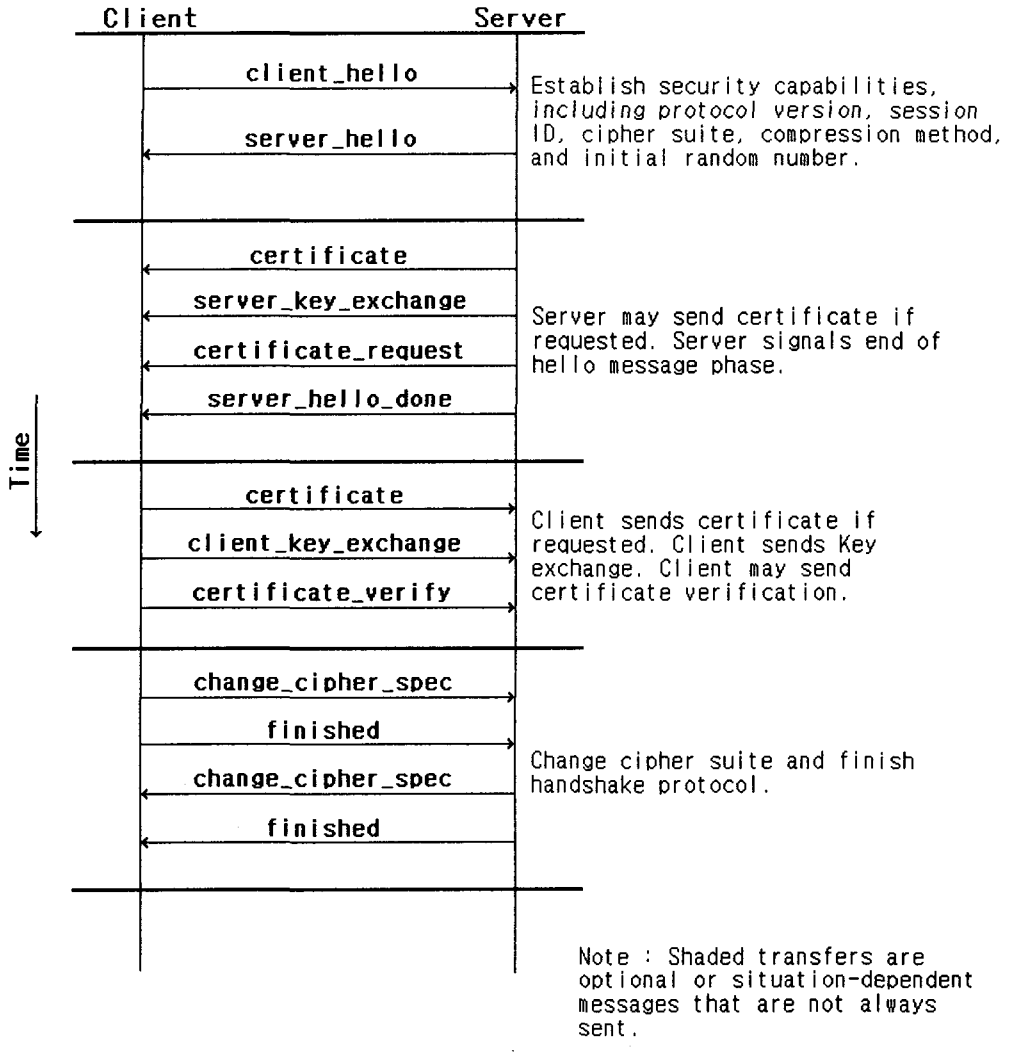


그림 6 Handshake Protocol Action

3.2.4 SSL Change Cipher Spec Protocol

서버와 클라이언트 서로가 보내게 되는데, 앞으로의 데이터전송에 서로 협의된 Cipher Spec을 따르겠다는 것을 알려주는 역할을 한다. Change Cipher Spec Protocol은 그림 7(a)와 같이 값 1을 갖는 single 메시지로 구성되어있고, 이것은 현재의 연결에서 사용할 Cipher Suite를 갱신하기 위해 사용된다. 즉, pending state의 정보를 current state에 복사한다.

3.2.5 SSL Alert Protocol

Alert Protocol의 구성은 그림7(b)와 같다. 첫 번째 byte는 warning과 fatal의 두 가지 level을 가지는데, 만약 level이 fatal인 경우에는 즉시 연결을 종료하게된다. 두 번째 byte는 Alert을 나타내는 code값을 가진다.[5][7]

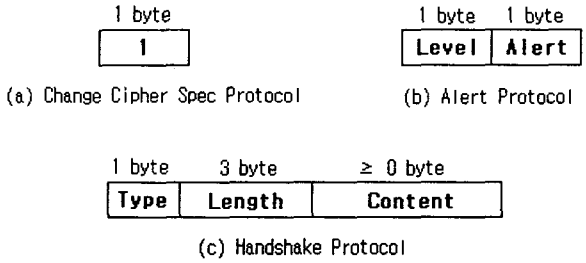


그림 7 SSL prtocol 구조

3.2.6 SSL과 TLS

1996년 SSLv3.0이 발표된 후, 1998년에는 SSLv3.0이 TLS (Transport Layer Security) 1.0으로 개선된 후 IETF(Internet Engineer Task Force)의 표준으로 추진되고 있다.

제 4 장 공개키와 SSL을 이용한 전자상거래시스템 구현

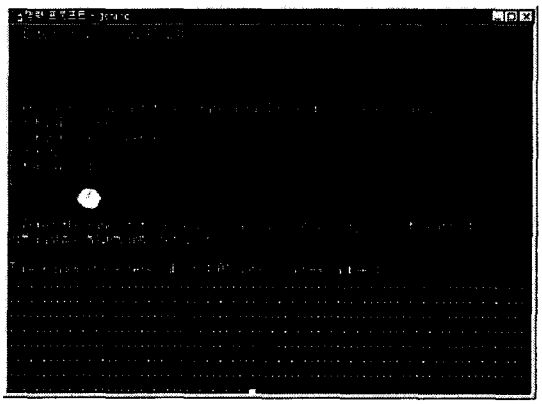
제 4장에서는 앞에서 언급한 내용을 토대로 공개키의 구현과 가상의 쇼핑 몰 구축, SSL의 설정 등에 대해 논의를 하고자 한다. 본 논문에서는 시스템 구현을 위해 WinNT 4.0 Server 환경에서 Oracle 8 DBMS와 Oracle Application Server v4.0.7을 사용하였다.

4.1 Oracle Application Server에서의 SSL 설정

클라이언트와 서버 사이의 SSL 설정을 위해서 먼저 클라이언트는 자체 인증서를 발급하고 이를 공인 인증기관으로부터 인증을 받는다. 그 후에 사용할 서버에 이를 설치함으로써 설정이 끝나게 되고, 이로써 서버와 클라이언트사이에 신뢰할 수 있는 채널이 생성된다. 이렇게 생성된 채널을 통하여 안전하게 통신을 할 수 있다. 본 논문에서는 공인 인증기관으로 Verisign을 선택하였다.

4.1.1 클라이언트 인증요청서 작성

① 인증서 작성을 위해 c:\orant\ows\4.0\bin\genreq.exe를 실행
② 설치과정에 따라 입력

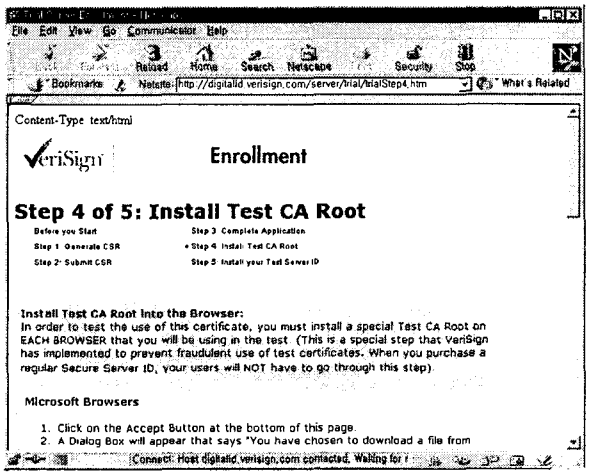



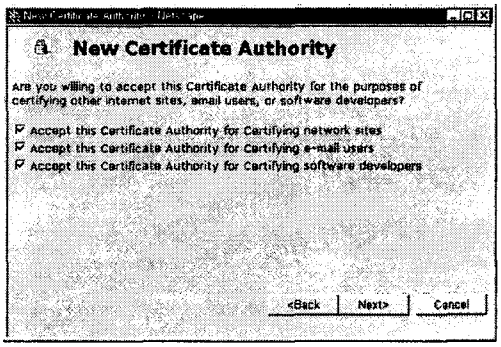
③ 현재의 디렉토리에서 servname.der, privkey.der, certreq.pkc가 생성되었는지 확인

4.1.2 인증기관을 통한 인증

① Web browser를 통해 인증기관(http://www.verisign.com)에 접속하여 web site security 선택

② Free Trial Web Server ID 접속
③ Enroll Now 선택
④ 설치단계에 따라 입력
⑤ trial certificate를 사용하므로 Test CA Root를 설치해야 한다

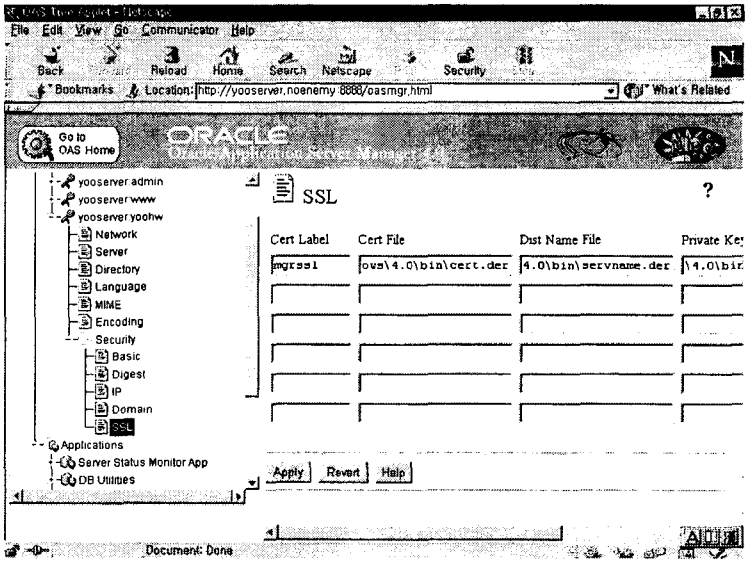




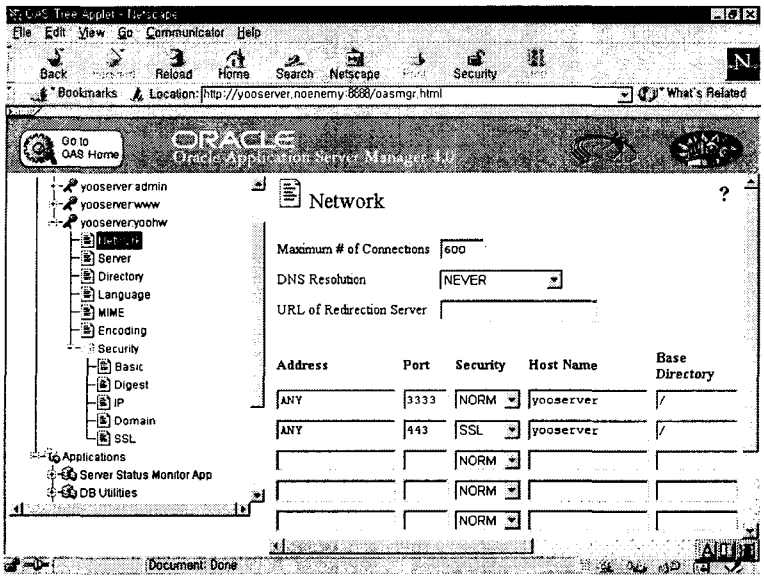
4.1.3 Oracle Application Server로의 인증서 설치

인증서는 OAS의 Listener Configuration을 통해 설치되는데, 이 인증서는 인증기관에서 사용자의 E-mail을 통해 오게된다. mail 중 ---BEGIN부분을 복사하여 cert.der이라는 파일로 만들어 작업하면 된다.

① Listener/Security/SSL에 인증기관으로부터 인증받은 인증서를 등록한다.



② Listener의 Network부분에 port 및 security등 해당 설정을 한다.



4.2 쇼핑몰에서의 SSL 적용

SSL을 적용하기 위해 Oracle DB와 Oracle Application Server를 이용하여 B to C 전자상거래에 해당하는 가상의 음반판매 쇼핑몰인 Music Club을 구축하였다.

■ SSL 적용 화면이다.

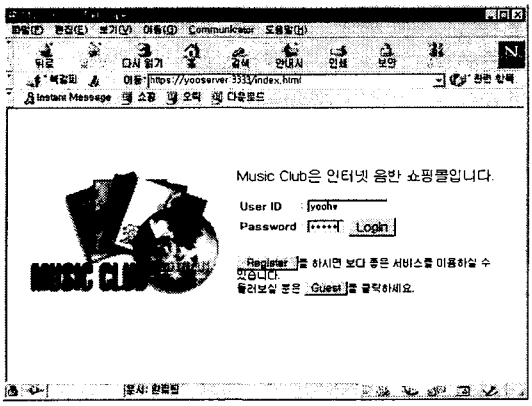


그림 8 로그인 화면

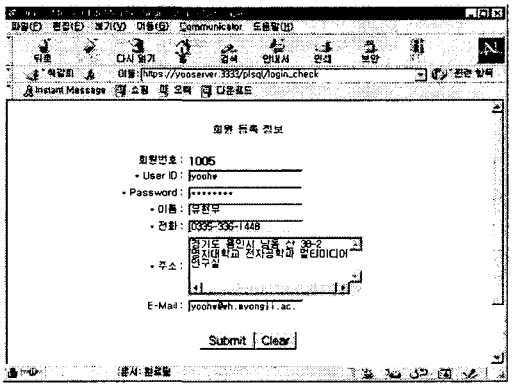


그림 9 사용자 등록

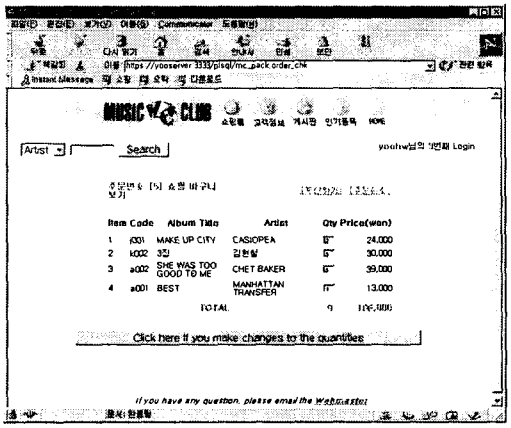


그림 10 주문하기

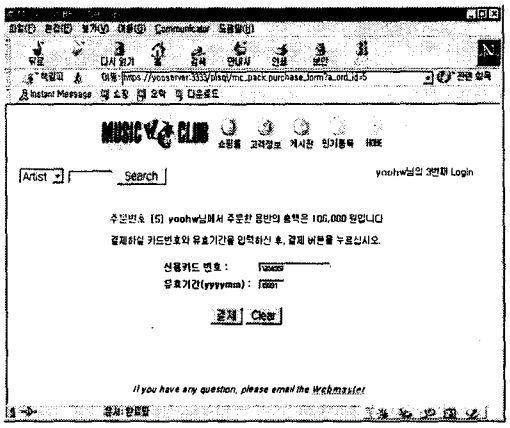


그림 11 계산과정

제 5 장 결 론

본 논문에서는 인터넷을 통하여 이루어지는 전자상거래에서 발생가능한 위협요소를 분석하였고 이의 대안으로 Network에서 보안문제를 해결할 수 있는 Security Socket Layer(SSL)를 구축하였다. 이것을 바탕으로 사용자 인증 뿐 아니라, 인터넷상에서 전자화폐의 전송이나, 사용자의 구매정보등 클라이언트와 웹서버간의 전송되는 정보를 보호할 수 있다.

향후 연구 방향은 대칭키방식이나 공개키방식의 암호화 알고리즘을 추가하여 전자상거래의 구현에 적용하는 것이다. SSL과 RSA의 이중 보안으로 인해 제 3자에 의한 정보 공격이 더욱 어려워 질 것이다.

이러한 연구의 노력은 인터넷을 사용하는 사용자에게 신뢰를 줄 것이며, 아울러 많은 사용자들이 보안성이 강화된 전자상거래를 이용하게 되리라 생각된다.

제 6 장 참고문헌

[1] 박창섭, 암호이론과 보안, 대영사, pp.261-303, 1999

[2] 동성정보통신, http://www.icash.co.kr

[3] 이니시스, "이니페이", http://www.inicis.com

[4] 전자신문사, http://www.etnews.co.kr/etnews

[5] http://home.netscape.com/eng/ssl3/ssl-toc.html

[6] Lincoln D.Stein, *Web Security*, Addison Wesley, pp36-43, 1998

[7] William Stallings, *Cryptography and Network Security principles and practice 2nd*, pp.355-473, 1999

[8] "미들웨어를 이용한 전자상거래 플랫폼 구현방안에 관한 연구", 한국전산원연구보고서, 1996

[9] "전자상거래 산업기술 동향", 정보처리학회지 vol.6 no.1, 1999