工學碩士 學位論文

전자상거래 시스템을 위한 통합인증서버의 구축 및 SSL의 적용

指導敎授 金 錫 泰

이 論文을 工學碩士 學位論文으로 提出함

2004年 8月

釜慶大學校 産業大學院

情報通信工學科

曺 容 雄

이 論文을 曺容雄의 工學碩士
學位論文을 認准함

2004년 6月

主 審 工學博士 張 珠 錫 (印)

委 員 工學博士 朴 奎 七 (印)

委 員 工學博士 金 錫 泰 (印)

목 차

Construction of Integrated Certification Server and Application of SSL to Electronic Commercial System

Department of Information Communication engineering Cho, Yong-oong
Directed by Professor Kim, Seok-tae

On the Internet, a construction of integrated certification server that enables both Electronic Commerce(EC) & Public Relations(PR) of a company is generalized. In order to construct the server, essentially, Secure Socket Layer(SSL) in enforcing security to a public key algorithm for constructing the server has to be applied.

In this paper, the new strategy of applying SSL to construct certification server and issuing certification signature is proposed. In addition, We present in detail how SSL services are composed of, and how to transmit keys and keep security and finally apply an encryption algorithm, SSL services and public keys. Consequently, Win2k server equipped with the certification server is constructed to verify the usability of the proposed mechanism in this paper. Thereafter, We monitor the EC system of Sinhan Steel Co. in which the system is deployed.

The advantages of the system are self-issuing and renewalling certifications, the integrity of a certification server and a web server in the operating system.

1. 서론

최근 범국가적으로 인터넷 환경이 갖추어 지면서, 초기 B2C (Business to Customer) 에 머물렀던 것이 지금은 본격적인 기업간 전자상거래시스템으로 옮겨가고 있다.

기업간 전자상거래시스템은 웹이나 인터넷을 통해 이루어지는 모든 비즈니스 거래를 의미한다. 다시 말해서, 기업간에 통합적이고 자동화된 정보체계 환경 아래 전자적인 매체를 이용하여 일상적인 상거래 뿐 아니라 대 고객 마케팅, 광고, 조달서비스, 생산, 수송, 행정, 재무, 구매 등을 포함하는, 거래에 필요한 모든 제반 정보를 교환하는 방식을 말한다[1-3].

그러므로 이러한 전자상거래 시스템의 성공적인 수행과 기업주와 사용자들이 마음 놓고 사용할 수 있는 시스템으로의 완성을 위해서는 먼저 메시지 전달과정에서 관여하는 여러 가지 시스템에 대한 신뢰가 전제되어야 한다. 즉 사용자의 시스템에서부터 공급자나 결재 대행자의 시스템까지의 통신 보안이 이루어지지 않으면 사용자 인증 때 개인의 신상정보를 제공하기 어렵고, 지불 자체도 이루어지기 힘들기 때문에 사실상 거래 자체가 이루어지지 않는다. 이 때문에 전자상거래 분야에서 넷스케이프 사가 제안한 통신 보안 솔루션인 SSL(Secure Socket Layer)을 통하여 128bit 암호화 기술을 바탕으로 통신 데이터를 암호화함으로써 거래의 보안을 유지해왔다.

그러나 SSL 은 IP 를 기반으로 당사자들을 인증하고 보안 통신 채널을 생성하므로 IP 사칭 등에 의한 부정 거래를 막기 어려우며 다양한 네트웍 환경에 대응하기 어렵다는 단점을 가지고 있다. 이와 같은 단점을 보안하기

위한 통신 보안 솔루션으로 암호화기법[4 6]과 보안 프로토콜[7-8]이 제시되고 있다.

지금까지의 보안관련 연구로는 SSL 의 최대 단점인 느린 속도, 많은 계산량에 따른 서버 리소스의 저하를 극복하기 위한 서버 컴포넌트의 제작[9-10]또는 특정 목적을 위한 SSL을 적용한 전자 상거래 시스템의 모델[11-12]을 제안되었다. 하지만, SSL을 적용을 위한 컴포넌트 구현과 모델링도 중요한 부분이지만, SSL을 적용하기 위한 암호화 기술을 적용한 인증서와 인증서버에 관한 부분은 보안에 있어 무시될 수 없는 부분이다. 사용자 정보에 극히 민감한 전자 상거래의 서버/클라이언트 통신에 있어 데이터 서명 및 인증서는 필수적인 요소이며 이 인증 시스템 구축은 그 기반이 되기 때문이다.

본 논문에서는 전자상거래의 보안을 위한 통합인증서 생성과 통합인증서 서버를 구축하기 위하여 인증 시스템 모델을 이용한 단일 서버 구조의 통합 인증서버 시스템을 구축하고, SSL을 적용하는 새로운 방법을 제안한다. 또한 SSL 서비스는 어떠한 구조로 이루어져 있으며, 어떤 방식으로 키를 전달하고, 보안을 유지하는지를 확인하여 적용방법을 고찰한다. 그리고, 인증 시스템을 구축하기 위한 인증 시스템 모델링 기법을 살펴보고, 마지막으로 암호화 알고리즘과 SSL 서비스 및 공개키를 적용하는 방법을 살펴본다.

제안한 방법의 유용성을 확인하기 위해 Microsoft Windows 2000 서버를 기반으로 SSL을 적용하여 중소기업용 인증서버를 구축한다. 또한 결과를 검증하기 위해 (주)신한스틸 전자상거래 사이트를 구성하고 로그인 부분과 상품주문 및 결재, 견적서 작성 등을 실제 운용한다. 본 방법의 장점은 자

체 인증서 발급 및 갱신이 가능하고, 인증서버 및 웹 서버가 OS에 통합 운영되어 효율적이고 비용절감이 가능하다는 것이다.

2장에서는 전자상거래 보안을 위한 인증서 및 통합인증시스템 모델을 구현하기 위해 인정서 개요 및 기법과 보안 알고리즘과 인증기관 구성 모델에 대하여 기술한다. 제3장에서는 통합인증시스템을 구축하기 위하여 인증키 생성 및 발급과 인증서 발급 Process를 구현하고 보안이 강화된 SSL을 직접 도입한다. 그리고 제 4장에서는 2장과 3장을 기반으로 하여 (주)신한스틸 전자상거래 사이트를 구성하고 로그인 부분과 상품주문 및 결재, 견적서 작성 등을 실제 운용하여 검증한다.

2. 전자상거래 보안을 위한 인증서 및 인증시스템 모델

2.1 인증서 개요

인증서에는 네트워크 상에서 신원을 확인하는 데 사용되는 정보가 들어 있다. 인증서는 ID의 기본 형태와 비슷하며 연결하기 전에 웹 서버와 사용자가 서로를 인증하는데 사용한다. 또한 인증서에는 클라이언트와 서버 간의 SSL(Secure Sockets Layer) 연결 설정에 사용되는 키 또는 암호화 값이 들어 있어, 이 연결을 통해 보내진 신용 카드 번호 같은 정보는 암호화되므로 인증 받지 않은 조직 또는 사람이 훔쳐보거나 사용할 수 없다.

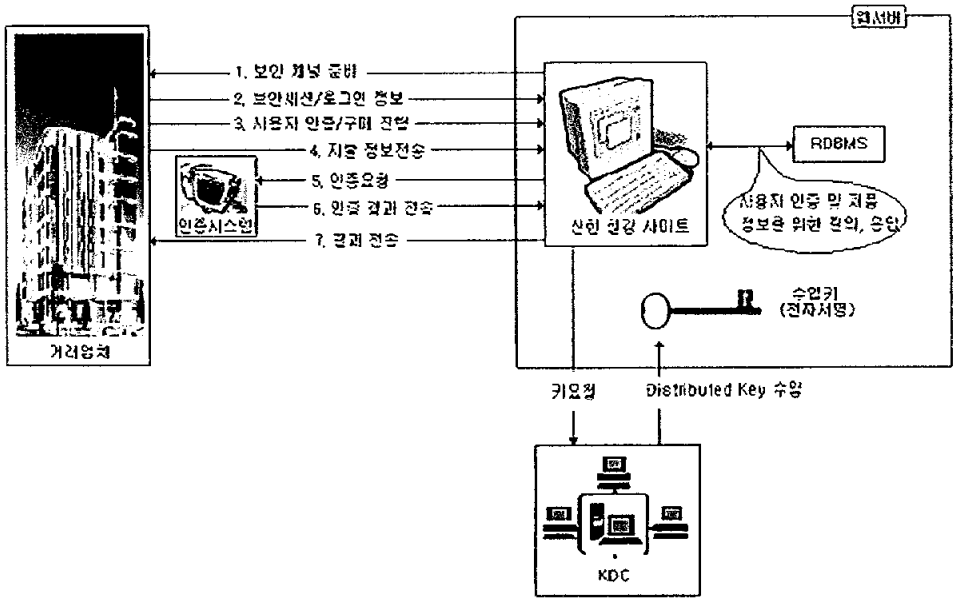


그림 2-1 인증시스템 예시

인증시스템의 예를 그림 2-1과 같이 나타낼 수 있다. 먼저 클라이언트들은 일단 세션을 요청하고 보안 채널을 요청 및 준비한다.(보안채널 준비) 다음 SSL을 거쳐 로그인 정보를 요구하게 된다. 여기서 SSL에 사용되는 인증서에는 두 가지 종류가 있다. 클라이언트 인증서에는 사이트로 액세스를 요청하는 클라이언트에 대한 개인 정보가 들어 있어, 이 정보로 클라이언트의 신원을 확실하게 확인한 다음 사이트에 액세스를 허용한다. 서버 인증서에는 서버에 대한 정보가 들어 있어, 클라이언트가 서버를 확실하게 확인한 다음 민감한 정보 등을 공유할 수 있도록 한다.(보안 세션/로그인 정보) 다음 사용자 인증 및 구매 과정을 거쳐 지불정보를 전송한다.(사용자 인증/구매 진행, 지불 전송정보), 그리고 전송후 인증결과를 요구하고 전송한다.(인증요청, 결과 전송), 마지막으로 결과를 전송후 세션을 닫는다.(전송결과) [13].

2.1.1 암호화 방식

암호화의 방식에는 대칭형(비밀키)과 비대칭형(또는 공개키)으로 크게 구분된다. 비대칭형 암호는 암호화키와 복호화키가 다르기 때문에 송신자와 수신자가 각각 다른 암호화키를 갖게 된다. 이중 암호화키는 인증기관의 서버에 공개하게 된다. 이 공개된 암호화키를 공개키라고 하고 이 방식은 공개키 구조방식(PKI:Public Key Infrastructure)하고 한다. 위의 것은 대칭형, 아래쪽은 비대칭형을 그림 2-2에서 도시하고 있다.

인터넷상에서 송신자와 수신자가 동일한 인증기관에 접속하고 있다면 임의의 송신자도 특정한 수신자에게 암호화된 텍스트를 보낼 수 있다. 즉, 그림 2-2와 같이 자신의 서명을 개인키로 암호화하고 상대방의 공개키로 암

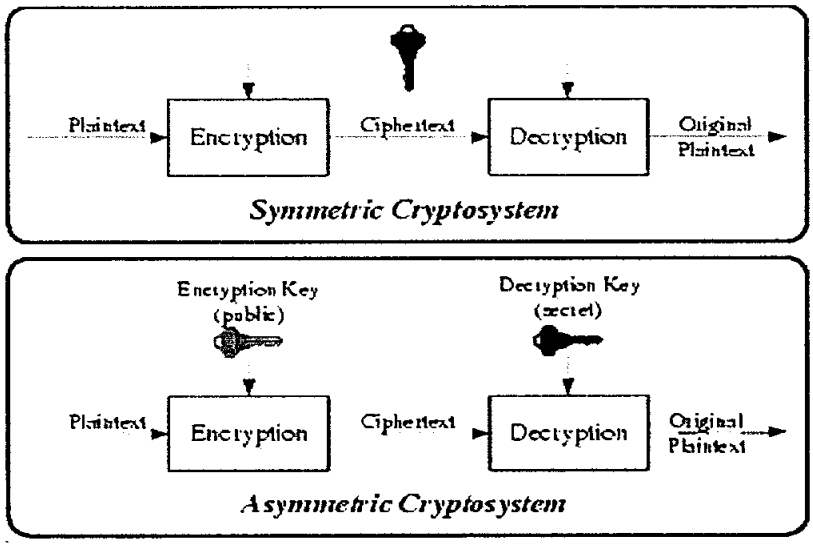


그림 2-2대칭형 암호화와 비대칭형 암호화

호화한 텍스트를 상대방에게 보내면 상대방은 상대방의 공개키로 상대의 서명이 진실한 것을 확인하고 자신의 개인키로 텍스트를 해독하게 된다.

이 방식은 공개키를 제외한 개인키는 오직 1명만이 알고 있으므로 유출의 위험이 적어지게 된다. 그러나 공개키에 의한 개인키의 역 산출이 될 위험이 있다. 그리고 반드시 송신자와 수신자가 공통의 인증기관의 서버에 접속하여야 하는 단점이 있다.

2.1.2 서버 인증서

서버 인증서는 웹 서버와 서버의 웹 컨텐트를 지원하는 조직에 대한 정보가 들어 있는 디지털 ID로서, 사용자가 서버를 인증하고 웹 컨텐트의 유효성을 확인하여 보안 연결을 만드는 데 사용된다. 또한 서버 인증서에는 클라이언트와 서버 간의 보안 연결을 만드는 데 사용되는 공개키가 들어

있다.

식별 방법으로서 서버 인증서의 성공은 사용자가 인증서에 포함된 정보의 유효성을 신뢰하는지 여부에 좌우된다. 예를 들어, 회사의 웹 사이트에 로그인하는 사용자는 회사의 서버 인증서의 내용을 보았더라도 신용카드 정보 제공을 주저할 수 있다. 특히 회사가 오래되지 않았거나 잘 알려지지 않은 경우에 더욱 신용카드로 계약하는데 망설여 질 수 있다. 이러한 이유로 인증서는 인증기관(CA)이라는 서로 신뢰하는 제 3의 기관에 의해 발행되거나 추천되는 경우가 있다. 인증기관의 주요 책임은 인증서를 찾는 사람들의 신원을 확인하여 인증서에 포함된 확인 정보의 유효성을 보증하는 것이다. 또한 웹 사이트 사용자와 사용자 기관의 관계에 따라 자신의 서버 인증서를 발행할 수 있다.

2.1.3 클라이언트 인증서

클라이언트 인증서는 클라이언트 정보가 들어 있는 전자 문서이다. 서버 인증서와 마찬가지로, 이 인증서에는 이 정보 뿐 아니라 IIS의 SSL 3.0 보안 기능도 포함된다. 서버 및 클라이언트 인증서에서 이러한 키 또는 암호화 코드는 인터넷 같이 열린 네트워크 상에 전송된 데이터를 암호화하고 암호를 해독하는 키 쌍을 형성한다. 일반 클라이언트 인증서에는 사용자 신원, 인증기관의 신원, 보안 통신을 위해 사용된 공개 키, 만기일과 일련 번호 등의 유효 정보가 포함된다. 보증 기관에서는 여러 형태의 클라이언트 인증서를 제공하며 그 인증서에는 필요한 인증 수준에 따라 다양한 정보가 들어 있다 [14].

2.1.4 세션 키

웹 서버에서는 근본적으로 같은 암호화 프로세스를 사용하여 사용자와의 통신 연결을 안전하게 한다. 안전한 연결을 설정한 후 웹 서버와 사용자의 웹 브라우저에서 정보를 암호화화거나 암호를 풀기 위해 특별한 세션키를 사용한다. 예를 들어, 인증된 사용자가 안전한 채널을 필요로 하는 웹 사이트에서 파일을 다운로드하려는 경우 웹 서버에서 세션키를 사용하여 그 파일 및 관련 HTTP 헤더를 암호화한다. 암호화된 파일을 받은 후 웹 브라우저에서는 같은 세션키의 사본을 사용하여 그 파일을 복원한다.

이 암호화 방법은 안전하기는 하지만 근본적으로 단점이 있다. 안전한 연결을 만드는 동안에 세션키의 사본이 보안이 이루어지지 않은 네트워크를 통해 전송될 수 있다. 즉, 컴퓨터 침입자가 마음만 먹으면 그 세션 키를 가로챌 수 있다는 의미가 된다. 그러나 그러한 가능성을 없애기 위해 웹 서버에서는 다른 암호화 방법을 추가로 사용한다. 이것이 공개 키 암호화이다.

2.1.5 공개 키 암호화

웹 서버의 SSL(Secure Sockets Layer) 보안 기능은 공개 키 암호화라고 하는 기법을 사용하여 세션키를 전송하는 동안 가로채지 못하게 보호한다. 공개 키 암호화는 개인키와 공개키를 사용하며 그 과정은 그림 2-3에 나타낸다.

그 순서로는 사용자의 웹 브라우저는 웹 서버와의 보안 통신 연결(https://)을 설정한다. 통신 보안에 사용할 암호화의 수준을 사용자의 웹

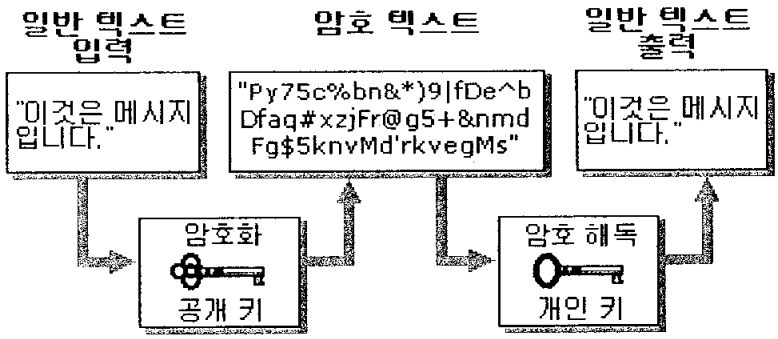


그림 2-3 공개 키 암호화 과정

브라우저와 서버가 함께 결정한다. 웹 서버가 브라우저에게 공개키를 보낸다. 웹 브라우저는 세션키를 만드는 데 사용된 정보를 서버의 공개키와 함께 암호화하여 서버로 보낸다. 서버에서는 개인키를 사용하여 메시지 암호를 풀고, 세션키를 만들어 공개키와 함께 암호화한 다음에 브라우저로 보낸다. 웹 서버와 브라우저에서는 모두 세션키를 사용하여 전송한 데이터를 암호화하거나 암호를 푼다.

2.2 인증서 기법 및 알고리즘

2.2.1 DES/3DES/AES

DES(Data Encryption Standard)는 Lucifer를 보완하여 IBM에서 개발한 블럭암호 알고리즘으로 1977년에 미국 표준국(U.S. National Bureau of Standards : NBS, NIST 전신)에서 표준으로 채택(FIPS PUB 46)되었다. 64비트 입력 블럭을 56비트 비밀키를 이용하여 암호화하는 블럭암호 알고리즘이며, 미국 연방정부의 데이터 보호용으로 출발하여 ANSI(American National Standards Institute)의 표준 암호 알고리즘, ABA(American

Bankers Association)에서 미국 내 금융 정보의 보호 표준으로 사용하기까지 사용 범위가 확대되어 현재 전 세계적으로 가장 널리 사용되고 있다. 그러나 56비트라는 짧은 키 길이로 인해 더 이상 안전하지 않다고 보는 것이 일반적인 견해이며, IPSEC이나 PKIX 등 새로운 응용들에서는 DES를 3회 반복되는 Triple DES를 사용하도록 권고하였으며, 최근 차세대 미국 표준인 AES가 확정됨에 따라 앞으로 점차적으로 그 사용범위가 줄어들 것으로 보인다.

DES 알고리즘의 플로우차트는 그림 2-4와 같이, DES암호에서의 Feistel 연산은 다음 XOR연산을 이용하며, 각 2배수(r) 비트의 블록을 적당한 라운드를 거쳐 암호문으로 변환한다. 그림에서 L0 와 R0는 각각 2배수 비트의 블록이 된다. 첫 입력에서 임의의 함수 f를 적용한 XOR 연산을 수행하고 각 블록을 치환한 후 다시 반복하여 r 라운드까지 반복한다. 그후 최종 치환후 암호문을 출력한다.

DES의 56비트라는 짧은 키 길이로 인한 안전성 문제를 해결하기 위한 대안으로 3개의 키로 DES를 3회 반복하여 사용하는 Triple DES를 사용하였다. 3DES는 속도가 DES보다 3배 정도 느리다는 단점에도 불구하고, 기존의 DES를 이용하여 쉽게 구현되며 DES의 안전성 문제를 해결하는 장점으로 인하여 여러 표준에서 사용되었다. 최근 차세대 미국 표준인 AES가 확정됨에 따라 앞으로 점차적으로 그 사용범위가 줄어들 것으로 보인다.

NIST는 3DES보다 더 효율적이고 안전하며 로얄티가 없어야 하는 등을 만족하는 알고리즘을 공모하고, 3년여에 걸쳐 15개의 후보 알고리즘을 공개적으로 평가하여, 최종적으로 AES 알고리즘을 선정/발표하였다. AES에 채택된 블럭암호는 Daemem과 Rijmen에 의해 개발되고 RIJNDAEL로 명명된 알고리즘으로 DES와 Triple DES를 대신해서 새로운 업계 표준으로 자리잡게 될 것이다.

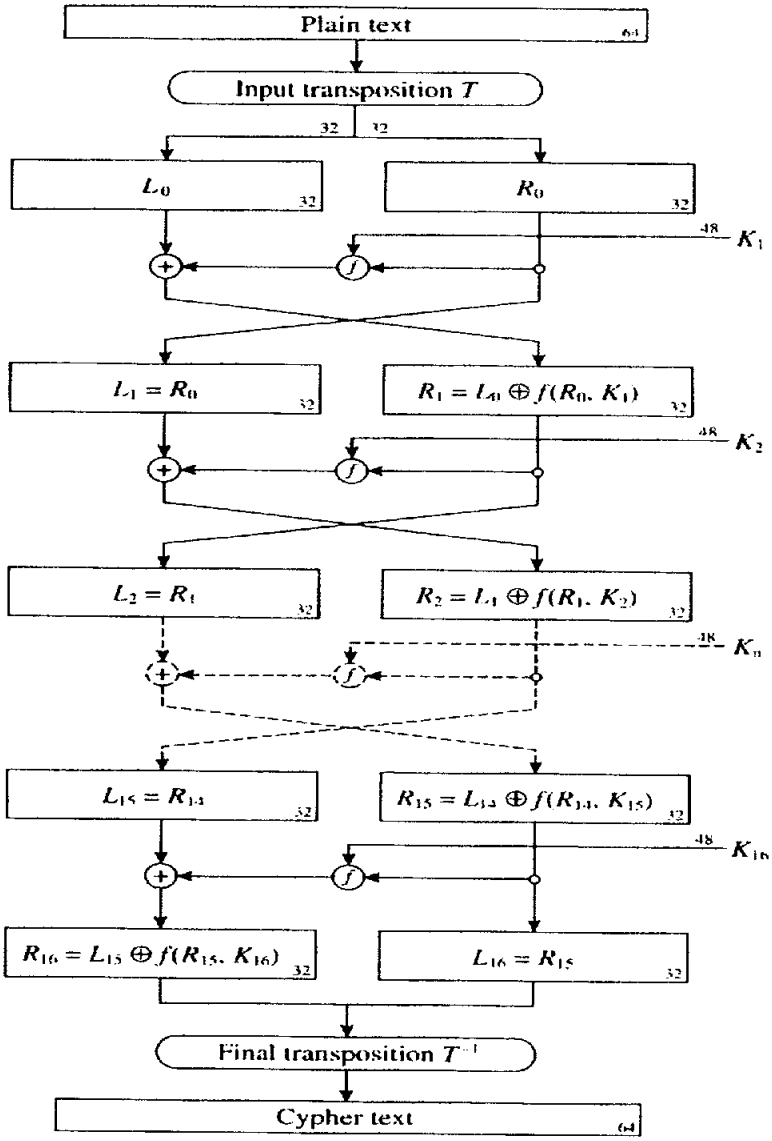


그림 2-4 DES 알고리즘 플로우차트

2.2.2 DSA/ECDSA

DSA(Digital Signature Algorithm)는 1991년 8월에 NIST(National Institution of Standard and Technology)가 DSS(Digital Signature Standard)에서 사용하기 위하여 발표한 정부용 전자서명 알고리즘으로 그 안전성은 이산대수 문제에 기반을 두었다. 그림 2-5에서 그 알고리즘을 도시하였다.

ECDSA는 DSA 전자서명을 타원곡선을 이용한 전자서명 알고리즘으로 변형한 것으로, 다른 공개키 시스템의 키 길이에 비해서 훨씬 짧은 키를 사용하여도 동일한 안전도를 제공하므로 스마트카드, 무선통신등과 같이 메모리와 처리능력이 제한된 분야에서 매우 효과적일 수 있다 [15].

DSA 알고리즘의 플로우차트를 그림 2-5에서 보여주고 있다. 왼쪽은 서명 생성, 오른쪽은 서명 검증작업을 위한 알고리즘이다.

서명 생성은 메시지 -> 보안 해쉬 알고리즘 적용 -> 메시지 다이제스트 생성 -> 개인키 적용 -> 디지털 서명 생성 의 과정을 거친다.

서명 검증은 "메시지 수신 -> 보안 해쉬알고리즘 적용 -> 메시지 다이제스트 생성 -> 디지털 서명과 공개키를 적용한 메시지 다이제스트와 비교 -> 검증 종료"의 과정으로 이루어진다.

2.2.3 IDEA

IDEA는 스위스에서 개발된 block cipher알고리즘의 하나로서 비밀키의 길이가 128 비트이고 입력 데이타의 길이는 64 비트이다. 비밀키의 길이가

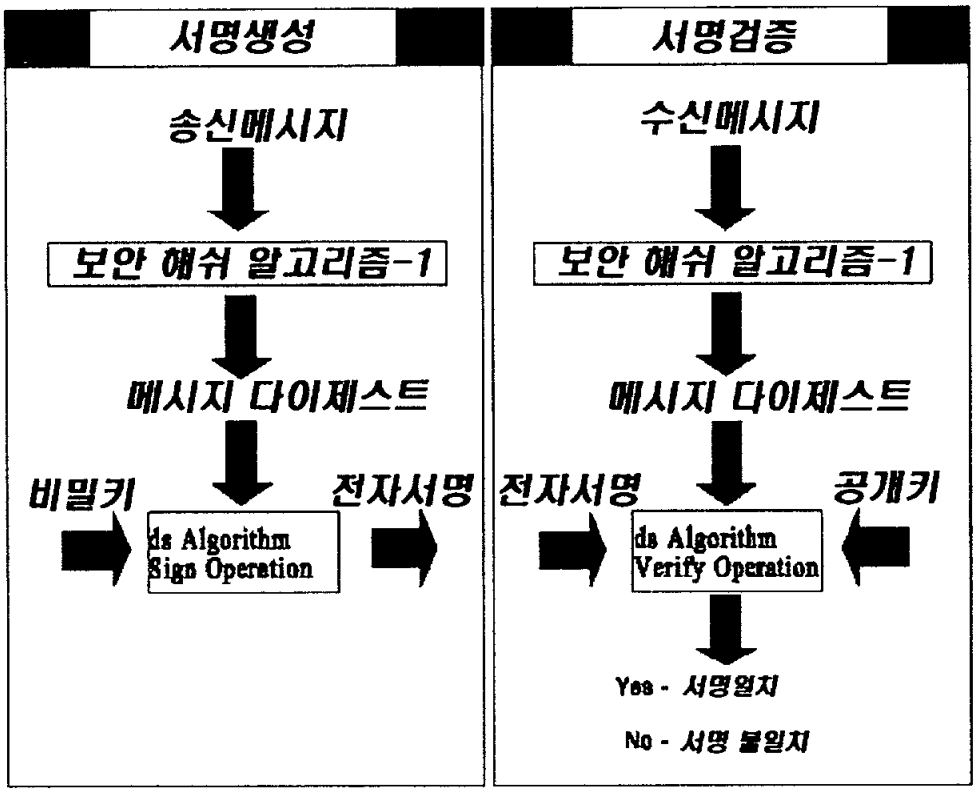


그림 2-5 DSA 알고리즘 플로우차트

128 비트이므로 매우 안전한 알고리즘으로 알려져 있다. 특히 IDEA는 differential cryptoanalysis에 매우 안전한 것으로 알려져 있다. 또한 알고리즘의 구조 자체가 소프트웨어뿐만이 아니라 하드웨어로도 구현이 매우 용이하게 되어 있다. 소프트웨어로 구현했을 때의 프로세싱 속도는 DES와 거의 비슷하다. 이는 비록 DES에서 채택한 비밀키의 길이가의 비밀키의 길이에 반밖에 되지 않지만 DES에서는 내부적으로 똑같은 연산을 16번 반복 수행하는데 비해 IDEA는 8번 수행하기 때문이다.

스위스에서 만들어진 IDEA(International Data Encryption Algorithm)는

초당 177Mbit의 처리가 가능한 빠른 암호화 방법이다. IDEA은 128-bit key, 8 round, 64-bit block cipher이며 주된 연산은 XOR, add mod 216, multiply mod 216+1이다. RSA와 더불어 PGP에 사용되는 방식이기도 하다.

2.2.4 Rijndael

정보의 기밀성을 유지하기 위한 블록 암호인 DES는 미국에서 연방 표준으로 제정된 이래 가장 널리 사용되어온 알고리즘이다. DES가 64 비트 블록에 56 비트 키를 사용하여 안전성에 문제가 제기되었던 바 AES는 향후 20~30년간 이용 가능한 안전성을 제공하여야 하며 128 비트의 블록 길이에 128, 196, 256 비트의 다양한 키 길이를 제공하여야 한다. 또한 DES의 후속 버전인 triple DES 보다 안전하고 고속이며 로열티가 없어야 한다는 조건으로 공모를 하게 되었다. 이에 미국 IBM의 MARS, 미국 RSA의 RC6, 역시 미국의 Twofish, 벨기에의 Rijndael, 영국, 이스라엘, 덴마크 연구자의 합작인 Serpent가 최종 라운드에 들어가는 알고리즘으로 선정되었다. 최종 라운드에서는 공개적으로 암호학적 안전성 분석을 하였으며 이 중 벨기에 Rijmen과 Daemen의 Rijndael 알고리즘이 최종 AES로 선정되었다. AES 선정을 위한 평가의 기준은 각 알고리즘이 안전한지, 효율적으로 구현되었는지, 유연성을 가지고 있는지에 중점을 두어 이루어졌다.

이의 결과로 Rijndael은 미국 연방 정부의 표준으로 제정되는 단계에 있으며 국제 사회에서 미국의 영향력이나 DES의 사용 전력으로 볼 때 앞으로의 국제 사회에서 가장 많이 사용되는 블록 알고리즘이 될 것으로 보인다.

2.2.5 SEED

SEED는 데이터 처리단위가 8, 16, 32비트 모두 가능한 블록 암호방식으로 되어있다. 입·출력문의 크기 및 입력키의 크기는 128비트로 되어있다. 또한 DC/LC에 대하여 안전하도록 설계되었다.

구조는 Feistel 구조로 내부함수는 SPN 구조이며, 비선형함수를 Look-up 테이블로 변형하여 사용한다. 키 생성 알고리즘은 알고리즘의 라운드 동작과 동시에 암·복호화 라운드 키가 생성될 수 있도록 설계되어 비대칭 암호화를 실시한다.

SEED는 Higher Order DC에 강하기 위하여 대수적 차수가 3이상인 부울함수를 사용하였으며, Related Key Attack에 강하기 위하여 Key Schedule에 비선형 함수를 사용한다. S/W로 구현시 3중 DES보다 고속이다 [16].

2.2.6 RSA(Rivest-Shamir-Adleman)

RSA 암호 알고리즘은 개발자의 이름 Rivest , Shamir, Adleman 첫자를 따서 지은 것으로 지난 77년에 발표되었으며 지금까지 1억개 이상 카피된 것으로 알려졌다. 이 알고리즘은 MS 윈도우, 넷스케이프 브라우저를 비롯해 로터스 등 수백 개의 소프트웨어와 연동이 가능하며, ITU, ISO, ANSI, IEEE 등 국제기구에 암호표준으로 제안돼 있으며 SSL 의 알고리즘을 제공한다.

하지만 RSA는 공개키 암호방식의 대명사로서 거의 모든 분야에 응용돼 왔으나, 몇 가지 단점이 있다. 먼저 막대한 계산량을 들 수 있다. 비트 수에

따라 다르겠지만 펜티엄급 컴퓨터에서 공개키와 비밀키를 만들려면 짧게는 20 여초, 길게는 몇 분까지 기다려야 한다. 복호화에도 많은 계산량이 요구되고 있어 컴퓨팅 파워가 떨어지는 휴대용 단말기에서는 사용하기 어렵다. 이런 문제를 해결하기 위해 등장한 암호알고리즘이 바로 타원곡선 알고리즘(Elliptic Curve)이다. 지난 해 1월 RSA사는 타원곡선 알고리즘을 RSA보다 한 단계 진일보한 공개키 알고리즘으로 발표한 바 있다.

RSA 시스템의 동작원리는 간단히 말해, 두 개의 큰 소수(소수는 그 숫자와 1로만 나누어지는 수이다)들의 곱과, 추가 연산을 통해 하나는 공개키를 구성하고 또 하나는 개인키를 구성하는데 사용되는 두 세트의 수 체계를 유도하는 작업이 수반된다. 한번 그 키들이 만들어지면, 원래의 소수는 더 이상 중요하지 않으며, 버릴 수 있다. 공개 및 개인키 둘 모두는 암호화/복호화를 위해 필요하지만, 오직 개인키의 소유자만이 그것을 알 필요가 있다. RSA 시스템을 사용하면, 개인키는 절대로 인터넷을 통해 보내지지

표 2-1 RSA Algorithm 의 예

단계 1	큰 소수 p=7와 q=13 선택
단계 2	n = p X q = 91을 계산
단계 3	n보다 작은 소수 e=43을 선택 (단, e는 (p-1) X (q-1) = 72와 인수를 가지지 않음)
단계 4	(e * d) mod 72 = 1 를 만족하는 d=67을 계산
단계 5	공개키 (n,e) = (91,43), 개인키 d =67 (단, p와 q는 폐기)
단계 6	암호화 c = (m * e) mod n = 2 X 72 mod 91 = 53
단계 7	복호화 m = (c * d) mod n = 53 X 67 mod 91 = 2

않는다. 그 예를 표 2-1에 들었다.

개인키는 공개키에 의해 암호화된 텍스트를 복호화하는데 사용된다. 그러므로 만약 A가 메시지를 B에게 보낸다면, A는 중앙의 관리자로부터 B의 공개키를 찾은 다음, 그 공개키를 사용하여 B에게 보내는 메시지를 암호화할 수 있다. B의 메시지를 받아서, B의 개인키로 메시지를 복호화 하면된다. 프라이버시를 확실하게 하기 위해 메시지를 암호화하는 것 외에도, B는 B의 개인키를 사용하여 디지털 서명을 암호화해서 함께 보냄으로써, 받는 사람 입장에서는 그 메시지가 틀림없이 바로 B에게서 온 것임을 확신시킬 수 있다. A가 메시지를 받으면, A는 B의 공개키를 이용하여 그것을 복호화할 수 있다.

이러한 과정을 이해하도록 표 2-2로 도시하였다[17].

표 2 2 공개키를 이용한 복호화 방법 예

구 분	누구의 어떤 키를 사용하나?
암호화된 메시지를 보냄	수신자의 공개키
암호화된 서명을 보냄	발신자의 개인키
암호화된 메시지를 복호화	수신자의 개인키
암호화된 서명 (발신자 인증) 을 복호화	발신자의 공개키

2.3 인증 시스템 개요

인증서 계층 구조는 그림 2-6의 계층 구조에서 각 발급자 또는 인증기관은 자신이 발급하는 인증서에 자신의 개인키를 사용하여 서명한다. 인증기관가 갖는 키 쌍의 공개키 부분은 인증서에 첨부되며, 이 인증서는 상위의 인증기관이 발급한다. 이런 과정이 필요한 수준만큼 계속 반복되며, 각 인증기관은 자신이 발급한 인증서의 신뢰성을 보증한다. 하지만 이 과정은 결국에는 하나의 최 상위 인증기관을 필요로 하게 되는데, 이 최 상위 인증기관을 루트 인증기관이라고 한다. 계층 구조에서 루트 인증기관보다 상위에는 아무것도 없기 때문에 루트 인증기관에 대한 인증서의 진품성과 출처를 증명해줄 수 있는 기관은 없다. 대신에 루트 인증기관은 자신의 인증서에 서명하고 자신이 루트임을 주장한다.

자신이 자기 자신의 신원을 증명하는 루트 인증기관을 허용하는 것은 안전하지 않은 것이 분명하다. 루트 인증기관의 인증서는 루트 인증기관의 신

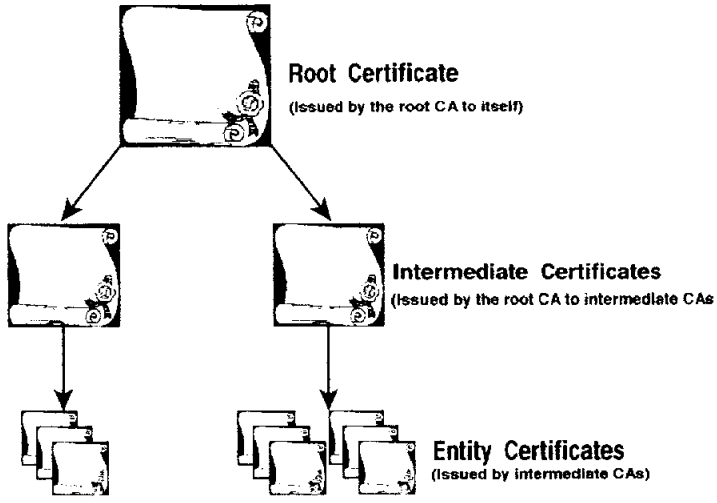


그림 2-6 인증서 계층구조

뢰 할 수 있는 공개 키 사본을 네트워크가 아닌 제 3자로부터 직접 받는 방식으로 입수하여(즉, out-of band 방식으로 입수하여), 그 키를 이용함으로써 검증된다. 루트 인증기관들도 공개 웹 사이트에서 공개 키 사본을 다운로드할 수 있게 제공한다. 일단 루트 키가 out-of-band 방식으로 배달되면 사용자는 루트 인증서의 진위를 검증하고 전체 인증서 체인의 진위도 검증할 수 있게 된다. 뿐만 아니라 각 인증서에 첨부된 디지털 서명은 인증서가 훼손되는 것을 방지하기 때문에 인증서 체인은 Internet처럼 보안 되지 않은 미디어에서도 자유롭게 전달될 수 있다 [18].

2.4 인증기관(CA) 구성모델

2.4.1 인증기관 모델

인증기관 계층구조모델을 선택은 PKI 솔루션을 구현할 때 아주 중요하다. 이는 각 모델마다 기술적, 행정적 장점 및 제약이 있으므로, 어떤 모델을 사용할 것인가를 이해하는 것은 PKI 도입을 계획하는데 있어 중요하다.

2.4.2 단독형 인증기관 모델

단독형 모델에서는 어떤 제3의 엔티티가 귀하의 조직에 대한 루트 키 및 인증서를 보유하며, 귀하의 사용자를 위한 모든 인증서를 발급하고 해지한다. 이 제3의 엔티티는 VeriSign, Thawte, Belsign, GTE Cybertrust 등과 같은 상업적인 인증기관 일수도 있고, 은행, 법률 회사, 무역 협회, 또는 인증서 발급 맡아주는 신뢰할 수 있는 다른 조직일 수 있다.

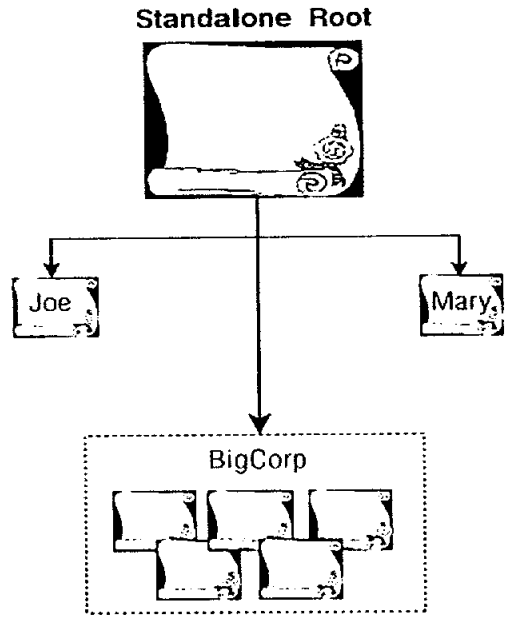


그림 2-7 단독형 인증기관 모델

단독형 인증기관 모델은 그림 2-7과 같이 조직 내/외 모두에서 신뢰를 받을 수 있게 하므로, 이 모델을 사용하면 보안 전자 메일과 전자 상거래 트랜잭션을 외부와 교환할 있다. 단독형 인증기관은 또한 귀하를 인증서의 발급, 해지 및 추적의 복잡성에서 해방시켜 준다. 그러나 단독형 인증기관은 인증서 관리에 관해 어떤 외부 엔터티를 신뢰할 것을 요구하고, 많은 제 3의 인증기관들이 발급된 각각의 인증서에 대해 개별적으로 요금을 징수할 수 있다.

2.4.3 엔터프라이즈 인증기관 모델

엔터프라이즈 인증기관 모델은 그림 2-8과 같이 기업은 자기 자신의 인

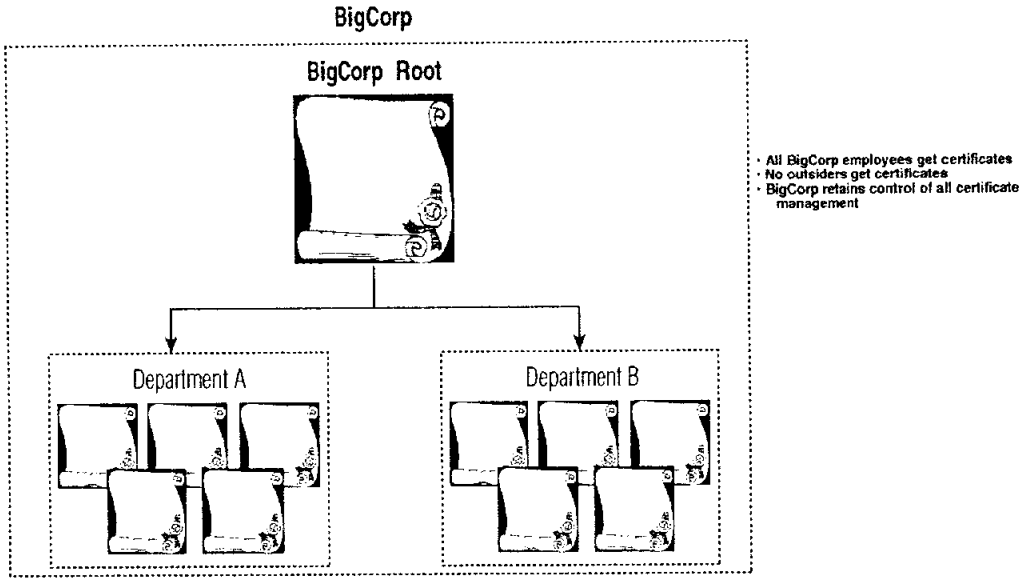


그림 2-8 엔터프라이즈 CA 모델

증기관으로 행동하며, 비즈니스 필요에 따라 인증서를 발급하고 해지한다. 외부 공급자가 관련되지 않으므로 귀하의 조직은 PKI에 대해 완전한 제어를 유지한다. 이러한 제어에 추가로, 기업 외부의 어떤 이도 귀하가 인증서를 발급해 주기 전에는 인증서를 가질 수 없음을 확신할 수 있다.

하위 인증기관이 있는 엔터프라이즈 인증기관은 조직의 개별 부서, 사업본부, 또는 자회사에 하위 인증기관을 추가함으로써 엔터프라이즈 인증기관 모델의 유연성을 확대할 수 있다. 예를 들어, 대부분 회사에서 관리직 직원은 일정 수준의 구매 권한을 가지고 있고, 높은 직위의 관리자일수록 큰 액수의 구매 권한을 가진다.

앞의 모델에서 각 하위 인증기관은 자신의 사용자를 위해 인증서를 발급한다. 회사의 루트 인증기관은 하위 인증기관에 인증서를 발급한다. 이러한 분할은 각 인증기관 소유자에게 자신의 사용자를 위한 인증서의 관리를 허용하면서 동시에 조직 수준에서 PKI의 전반에 대한 통제를 보존한다 [19].

3. 인증키 생성 및 발급을 위한 통합 인증서버 구축

3.1 시스템 구성

이론적으로는 System Root 인증기관과 하위 인증기관이 그림 3-1과 같이 다른 것으로 되어 있지만, 여기에선 웹 서버를 기준으로 하므로 독립적인 구조를 위해 Enterprise 인증기관으로 구성하며 웹 서버와 통합된 것으로 구성하였다.

루트에 설치된 System Root를 인증기관으로 하는 인증서버에서 인증서를 받아 IIS1, IIS2 에 설치하게 되며, IIS1의 가상 디렉토리에서 SSL 적용을 하게 된다. 각 하위 웹서버 (그림의 IIS Server 들), Root 인증기관에서

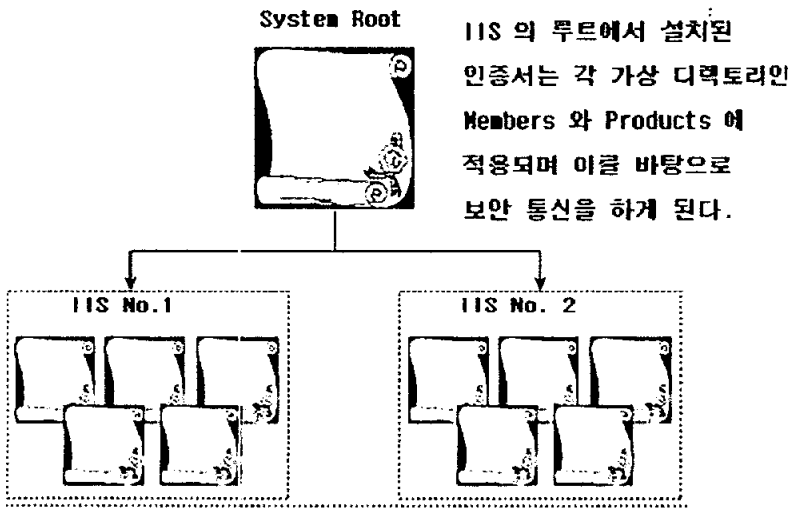


그림 3-1 시스템 구성도

인증서를 설치한 후에는 각 인증서는 루트키가 손상 및 해킹 되거나 소멸될때까지 유효하며, 또한 인증서의 유효기간이 다 되었을때에도 새로 갱신하고 받아야 한다. 만약, 각 하위 웹서버의 인증서가 손상되되었을 경우 루트에 재발급 요청을 하고 기존 인증서는 폐지한다.

3.2 인증서 발급 과정 구현

인증서는 Windows 2000 Server의 IIS(Internet Information Server)의 각 디렉토리마다 있는 보안 탭에서 따로 따로 설정해 주어야 한다. 지금 이 서버에는 기존 인증서가 설치되어 있으므로, 최초 발급 메뉴가 아닌 갱신 메뉴가 나와 있다. 하지만 그 과정은 큰 차이가 없으므로 이 과정을 대신하여 나타낸다.

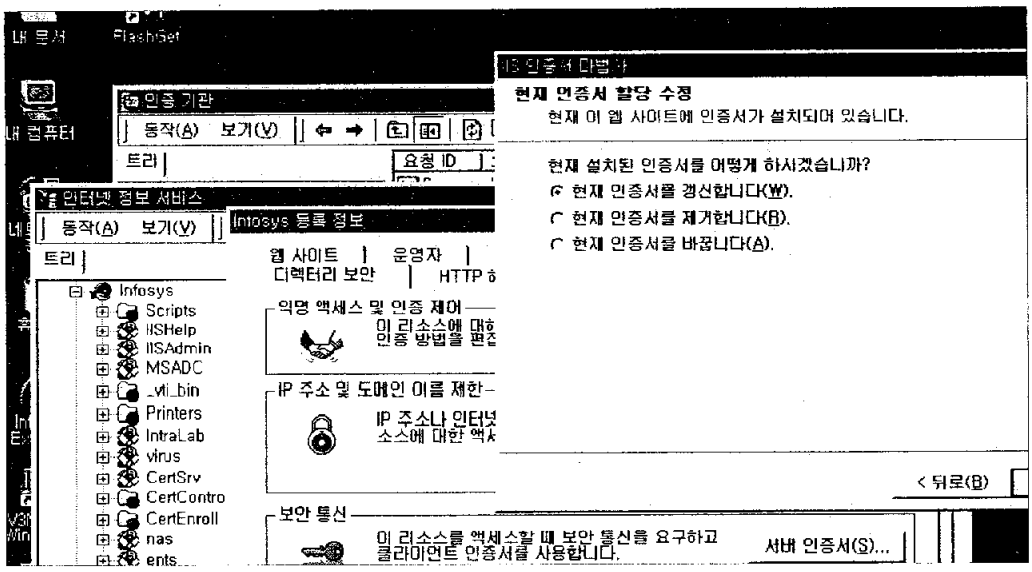


그림 3-2 인증서 발급 마법사 화면

기존인증서를 재발급 받기 위하여 인증서 갱신을 선택하는 화면은 그림 3-2로 인증 마법사에서 인증서를 요청하면, 요청을 즉시 보낼것인지, 나중에 보낼 것인지를 결정할 수 있다. 3rd Party Certification을 이용할 경우 즉시 요청을 하게 된다. 여기서는 Enterprise 인증기관을 예로 하였으므로 그림 3-3과 같이 요청을 지금 준비하여 나중에 보내거나 즉시 보낼 수 있다

인증키 생성 마법사가 만든 Key Request 파일은 그림 3-4와 같이 텍스트 형식으로 되어 있으며 이것을 복사하여 인증기관에 키를 요청하게 된다. 위와 같이 TXT File을 복사하여 3rd Party 인증기관에 인증서를 요청하게 된다. 지금 서버에 Certification Service가 설치되어 있으므로, 이것을 현재 서버에 적용하여 Key를 발급받을 것이다

Request 내용은 64 기수 Encoding 되어 있으며, ASCII 형식으로 되어 있

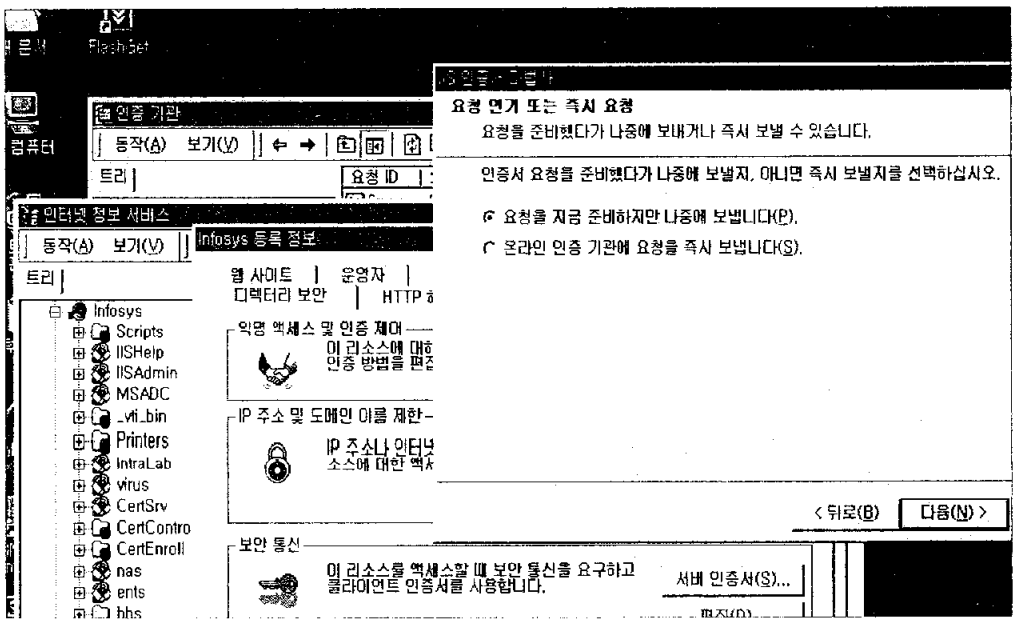


그림 3-3 발급 즉시요청/차후 요청 결정과정

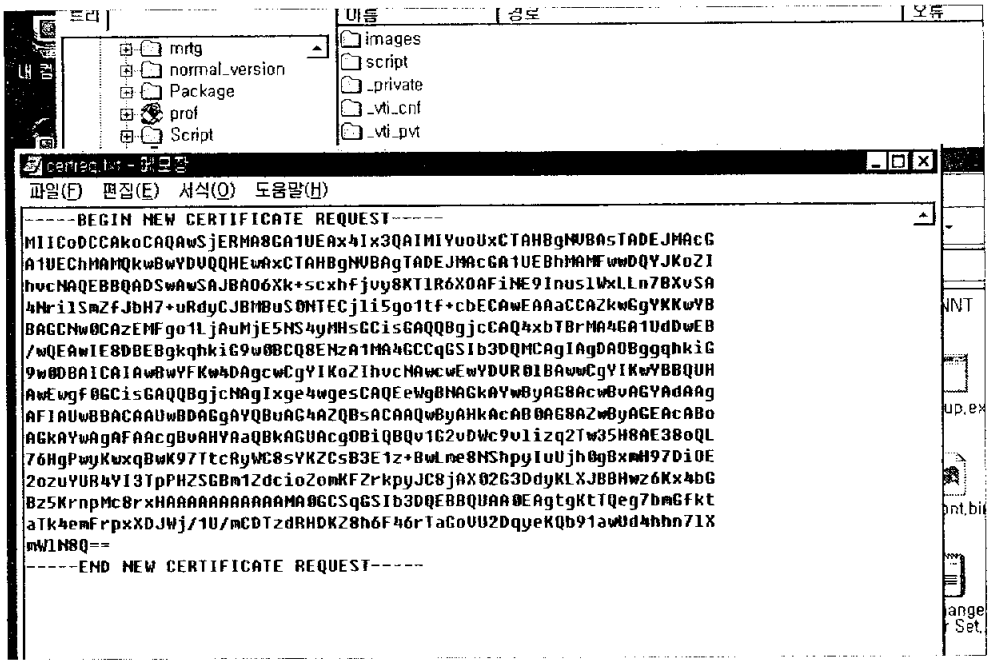


그림 3-4 생성된 Key Request txt File

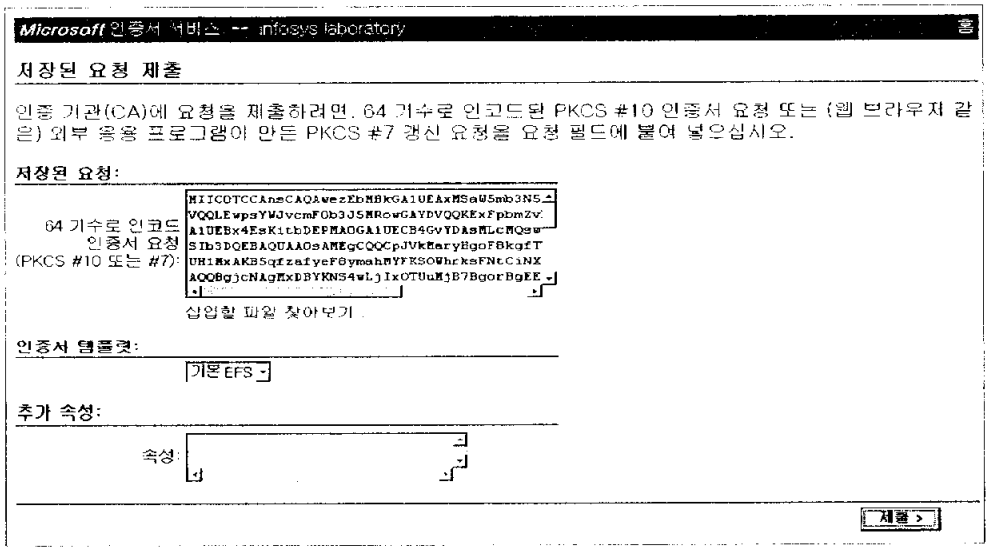


그림 3-5 인증서를 받기위해 Certsrv 디렉토리에 접근한 후 Request를 요청하는 화면

다. 이것을 복사하여 제출하게 된다.

인증 서비스를 이용해 인증서를 발급받기 위해서 IIS 의 기본 하위 디렉토리인 CertSrv 가상 사이트에 접속하였다. 인증서 서비스에 접속한 후 Request 내용을 제출하고 있는 화면은 그림 3-5에서와 같이 볼 수 있다.

고급 인증서 요청 화면은 그림 3-6과 같이 세 가지 옵션 중 앞에서 우리가 복사한 64 기수 인코딩 요청서를 제출할 것이므로 두 번째 옵션을 선택한다.

첫 번째 옵션은 3rd Party 인증기관에 제출 시 사용하는 옵션이다. 세 번째 옵션은 Smart Card 인증서를 요청하기 위한 Smart Card 등록 스테이션에 요구를 할 때 사용된다. PKCS #7(http://www.rsa.com/standards/)는 보안 정보가 있는 디지털 서명이나 디지털 봉투 등 암호화된 데이터 형식을 설명한다. 둘 다 IIS의 인증서 기능과 관련되며, PKCS #10

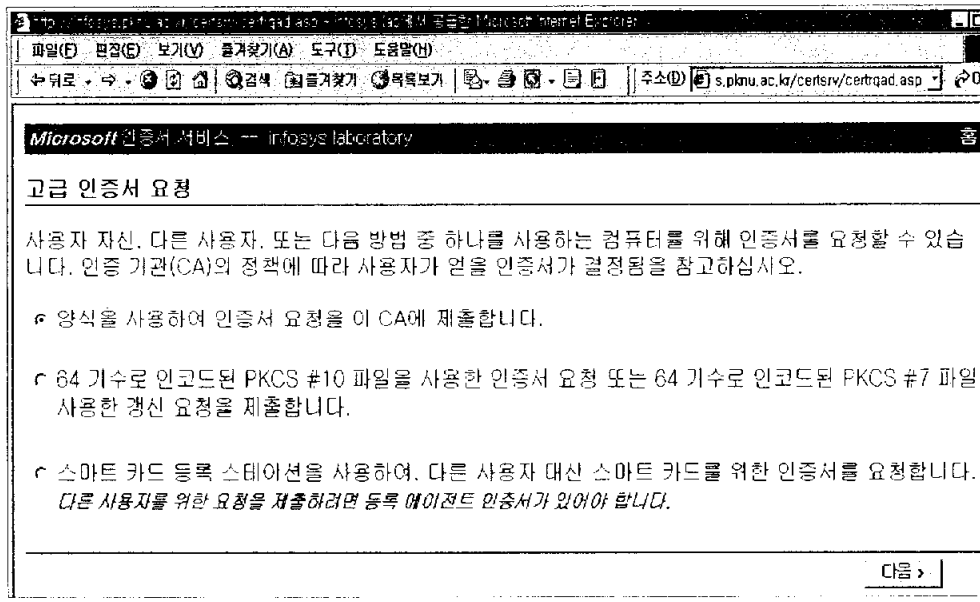


그림 3-6 고급 인증서 요청 옵션 선택

(http://www.rsa.com/standards/)는 인증기관에 제출한 인증서의 요청 형식을 설명한다.

인증서를 다운로드 받는 화면 그림 3-7이며, 최종으로 다운로드 받은 인증서는 웹을 인증하기 위한 해당 서버에 저장하여 IIS에 설치되게 된다. 필요한 경우(인증서를 직접 다운로드 시키기 곤란한 경우) 경로 다운로드를 통해 간접적으로 받아가도록 할 수 있다.

여기에서는 현재 인증서 요구서를 제출한 상태이므로 인증서를 설치해야 한다. IIS의 루트 항목에서 등록정보>디렉토리 보안>보안통신 항목을 선택하여 IIS 인증서 마법사를 실시한다. 그 첫 번째 단계 그림 3 8과 같이 나타 낼 수 있다. 이 과정은 보류중인 인증서 요청이 있을 때 이것을 어떻게 처리할 것인가를 선택하는 화면이다.

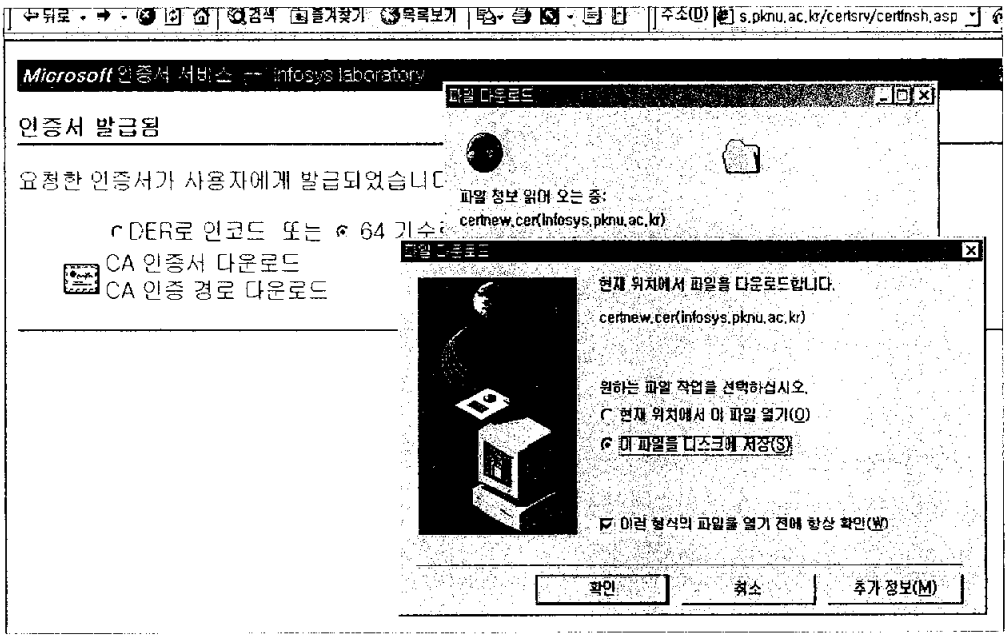


그림 3-7 인증서 다운로드

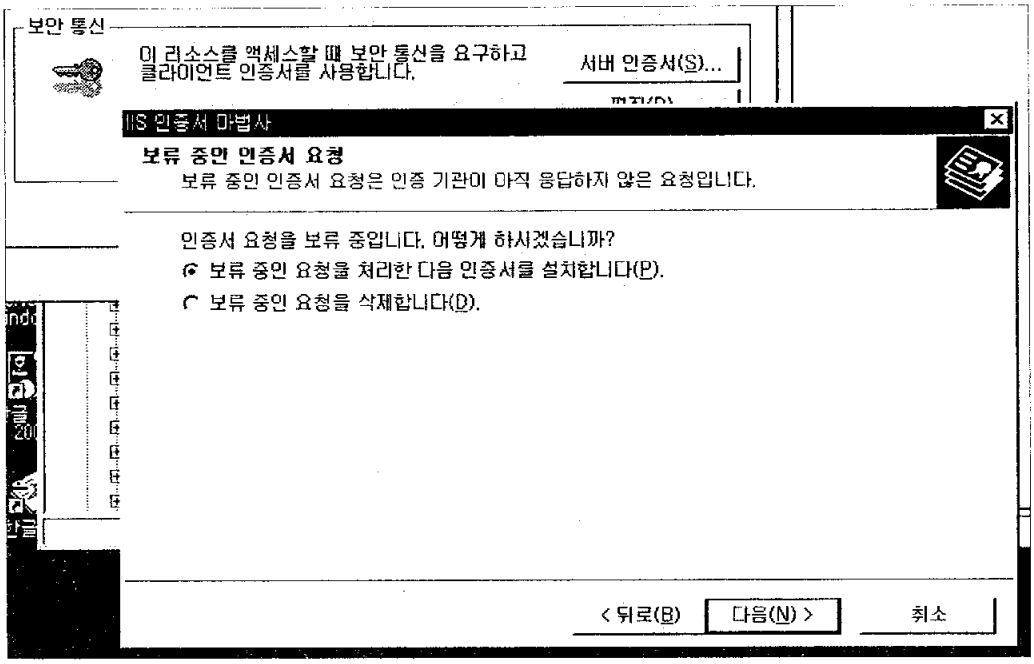


그림 3-8 보류중인 인증서 요청 처리 옵션 화면

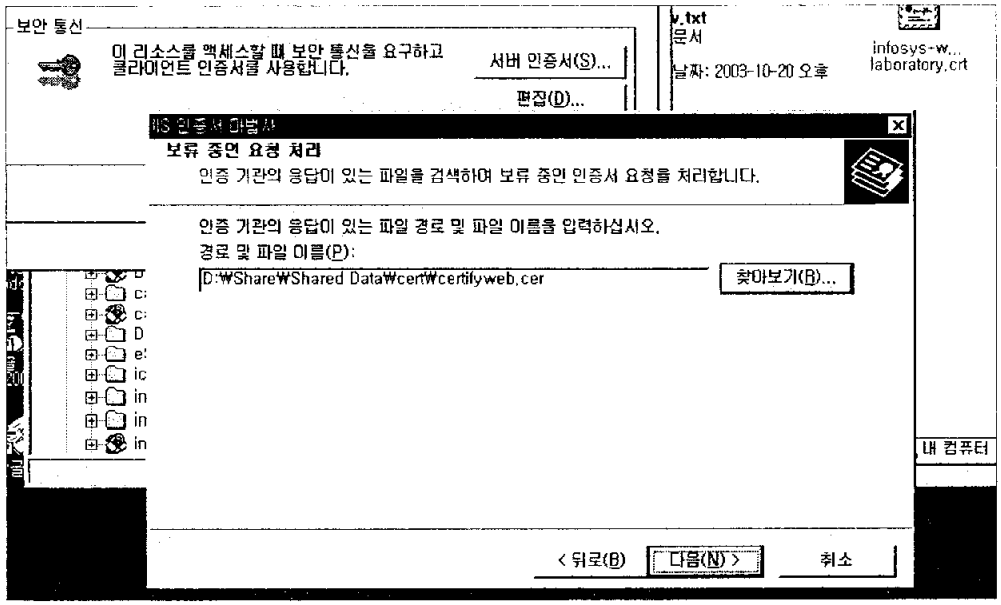


그림 3-9 보류중인 인증서 요청을 처리하기 위해 파일경로를 지정하는 화면

현재 기존 인정서 요청이 있었고 그것이 필요 없으면 두 번째 항목을 선택하면 기존 인증서 요청이 삭제된다.

또한 해당 서버에 저장된 인증서를 찾아 경로를 지정하여 그 서버에 반환하는 화면은 그림 3-9와 같으며, 마지막으로 인증서를 설치하기 전 인증서 요약정보를 그림 3-10에서 보여주고 있다. 또한 최종 인증서 확인한 화면으로 IIS를 설치 후 웹 루트 디렉토리의 등록정보에서 해당 인증서가 다음과 같이 인증서가 잘 설치되었음을 그림 3-11에서 확인 할 수 있다.

지금까지 인증서를 요청 및 발급 그리고 설치하는 과정을 보였다.

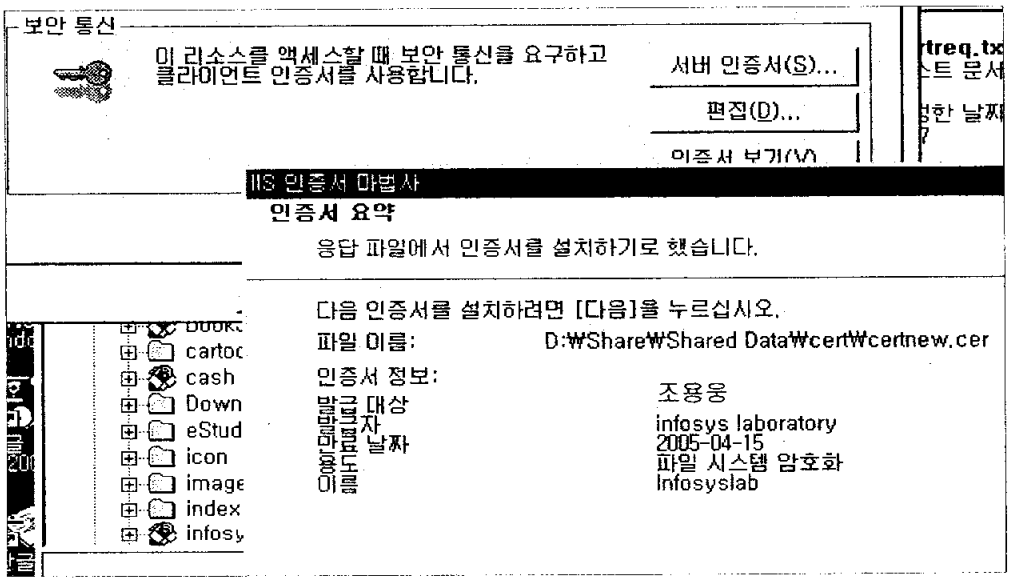


그림 3-10 해당서버에 저장된 인증서의 요약 정보 표시 화면

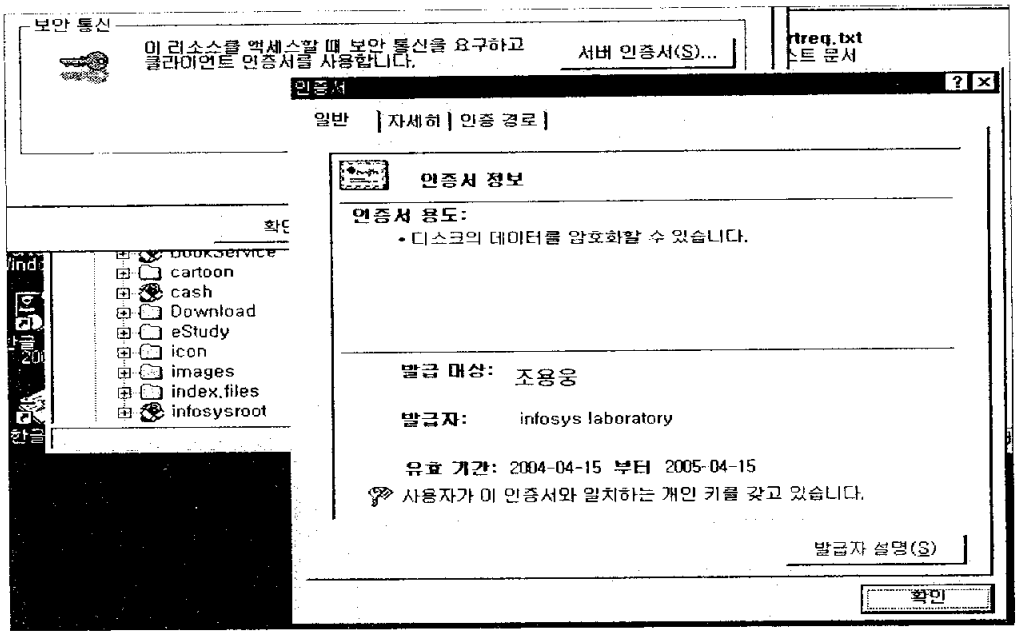


그림 3-11 IIS 서버에 인증서 확인화면

3.3 SSL(Secure Socket Layer) 개요

3.3.1 OSI 7 Layers

OSI는 통신 네트웍으로 구성된 컴퓨터가 어떻게 데이터를 전송할 것인가에 대한 표준규약 또는 참조 모델이다. 그 목적은 통신 제품을 만들 때 다른 제품과 모순됨이 없이 통신하도록 유도하는 것이다. 이 참조 모델은 통신의 종단에서 이루어지는 기능을 7 계층으로 정의했다. OSI가 잘 정의된 계층마다 관련된 기능을 따르도록 강하게 고수하지 않아도, 대부분의 제품들은 OSI 모델에 관련된 정의들을 따르기 위해 노력한다. OSI 모델은 또한 모든 사람이 동일한 관점에서 통신에 대해 교육하고, 논의하는 유일한 참조 모델로서 중요한 가치가 있다.

OSI 7 계층 구조는 그림 3-12와 같이 7개 항목으로 구성되어 있다. 여기에서 우리가 구축하고자 하는 SSL이 작동하는 부분은 그림의 Session Layer에 해당한다. Session Layer의 역할은 두 터미널 간의 연결 정보에 관한 부분을 처리하는 것인데, SSL은 Session Layer에 머물면서 TCP 계층(그림의 Transport 계층)에서 올라온 데이터를 해석하여 HTTP/FTP/SMTP 등의 계층(그림의 Application / Presentation 계층에 해당)에 올려주고 반대로 암호화 하여 내려주는 역할을 한다[20].

3.3.2. SSL (Secure Socket Layer)

SSL은 넷스케이프사에서 개발된 프로토콜로 특정시스템에서 사용되는 프로토콜이 아닌 일반 인터넷 환경에서 사용할 수 있는 프로토콜로 개발되었다. SSL 프로토콜은 웹 응용 프로그램과 TCP/IP 프로그램 사이에 존

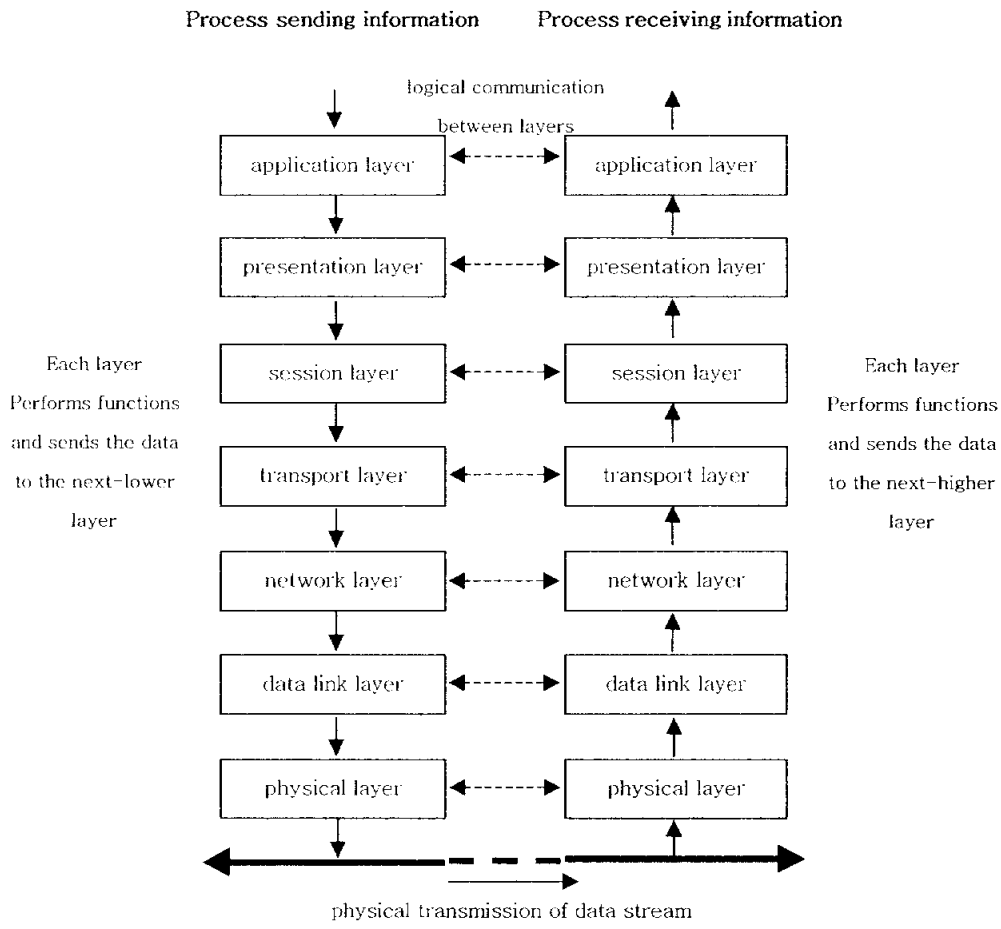


그림 3-12 OSI 7계층 구조

재하는 프로토콜이기 때문에 웹을 위한 HTTP 프로토콜뿐만 아니라 기존 인터넷응용프로그램에도 쉽게 적용 가능하다. 또한 암호화를 사용한 보안 방법으로 사용자가 신용 카드 번호나 전화 번호 같은 개인 정보를 서버와 교환하는 데 사용할 수 있다. SSL은 웹 내용의 인증을 확인하고 웹 사이트에 액세스하는 사용자의 ID를 선택적으로 확인한다.

인증서에는 SSL 보안 연결을 구축하는 데 사용된 키가 있다. 키는 SSL

연결 시 서버와 클라이언트를 인증하는 데 사용되는 고유한 값인데, 공개키와 개인키는 SSL 키 쌍을 형성합니다. 웹 서버는 키 쌍을 사용하여 사용자의 웹 브라우저와 보안 연결을 조정하고 보안 통신에 필요한 암호화 수준을 결정한다.

이 연결 형식을 설정하려면 웹 서버와 사용자의 브라우저 모두가 암호화 기능과 암호 해석 기능을 갖추고 있어야 한다. 암호화를 교환하는 동안, 즉 세션 동안 키가 만들어지고, 서버와 웹 브라우저 모두 세션키를 사용하여 전송된 정보를 암호화하고 암호를 해독한다. 세션키의 암호화 정도, 즉 강도는 비트 단위로 측정된다. 세션키를 구성하는 비트의 수가 크면 클수록 암호화와 보안의 수준이 높다. 이러한 높은 강도의 암호화키가 높은 보안수준을 제공하더라도 이를 구현하려면 많은 서버 리소스가 필요한데, 웹 서버의 세션키는 길이가 대개 40비트지만, 필요한 보안 수준에 따라 SSL은 네트웍 내에서 메시지 전송의 안전을 관리하기 위해 넷스케이프에 의해 만들어진 프로그램 계층이다. 넷스케이프의 생각은, 비밀이 보장되어야하는 메시지를 맡은 프로그램은 웹 브라우저 또는 HTTP와 같은 응용프로그램과, 인터넷의 TCP/IP 계층 사이에 들어가야 한다.

각 보안 프로토콜의 위치를 그림 3-13과 같이 도시하고 있다. 가장 왼쪽의 그림에는 SSL의 위치가 나타나 있다. 여기서 "소켓"이라는 용어는 데이터를 네트웍상의 클라이언트와 서버 프로그램 사이, 또는 같은 컴퓨터의 프로그램 계층끼리 주고받는 소켓 방식을 줄여서 말한 것이다. 넷스케이프의 SSL은 디지털 증명의 사용에도 포함되는 RSA의 공개/개인키 암호화 시스템을 사용한다.

SSL은 넷스케이프 브라우저의 없어서는 안될 핵심 부분이다. 만약 어떤

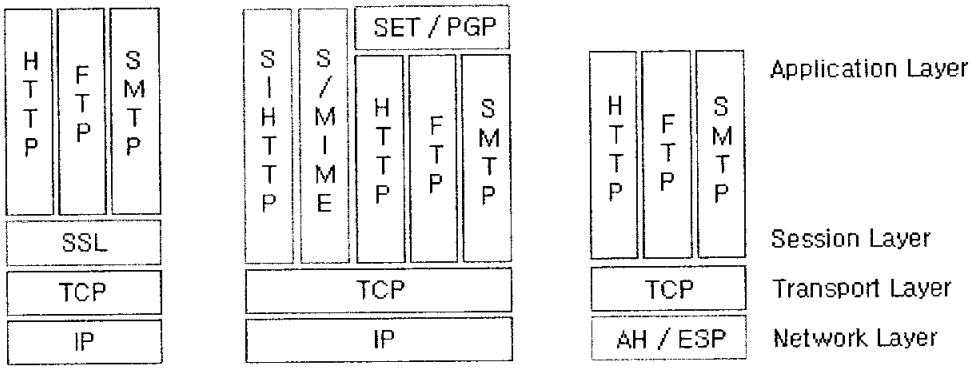


그림 3-13 각 보안 프로토콜의 개념도(SSL, SHTTP, IPSec)

웹사이트에 넷스케이프 서버가 설치되어 있으면, SSL을 사용하여 특정한 웹페이지가 SSL 액세스를 필요로 하는지를 식별할 수 있다.

넷스케이프는 SSL을 W3C에 표준 프로토콜로서 제안했으며, IETF는 웹 브라우저와 서버를 위한 표준 보안 접근방법으로 제시하였고, 이것이 표준화가 되었으며, 28비트가 될 수도 있다[21].

3.4 SSL의 구현

OS 는 Microsoft® Windows 2000 Server® 로 한다. 그리고 웹 서버는 운영체제 내에 있는 IIS (Internet Information Server ; Windows 2000 기본 웹 서버)를 이용하여 웹 서버와 운영체제 간의 연동을 꾀한다. IIS에 인증서를 수입하여 설치한다. 그리고 서버에서 제공하는 인증서 서버 서비스를 동시에 구현한다. 이로서 SSL을 구현하여 클라이언트의 로그인과 더불어 결재 시 보안을 구현한다.

보안키 수입개념도 그림 3-14과 같이, 3rd Party 업체의 인증서를 수입하

여 쓰는 경우를 도시하였다. 이때의 경우, 3rd 업체의 인증서를 비대칭 키의 형태로 받아와서 IIS 서버에 설치하여 사용하게 된다. 이때의 3rd Party 업체는 자신이 root 일수도 있고, root에서 분배된 키를 재분배하는 중간 분배자일 수 있다. 그러나, Windows Server 는 자체적으로 Certification Server 기능을 제공하며, 이를 이용하면 자신의 도메인 내에서 IIS 에서는 Security Key를 수입하여 설치 후 SSL을 구현할 수 있게 되어있다.

인증서 서버와 그 목록은 그림 3-15에 도시하였다. OS에서 Certification Server를 통해 발급받은 Key를 발급받은 뒤 IIS에 설치하였다. 그 후 Web Server Property를 통해 SSL을 세팅하고, https 프로토콜을 설치하고, 세팅하였다. SSL은 443번 포트를 통해 통신한다.

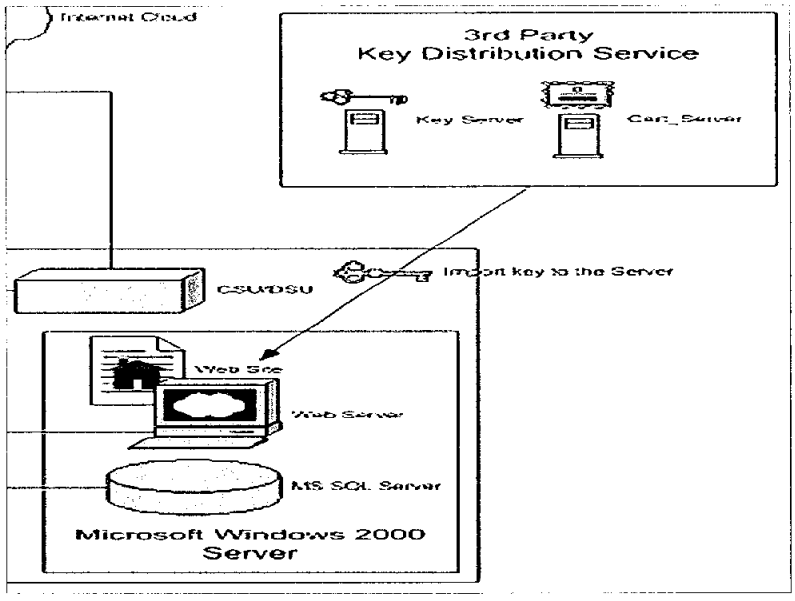


그림 3-14 보안 키 수입 개념도

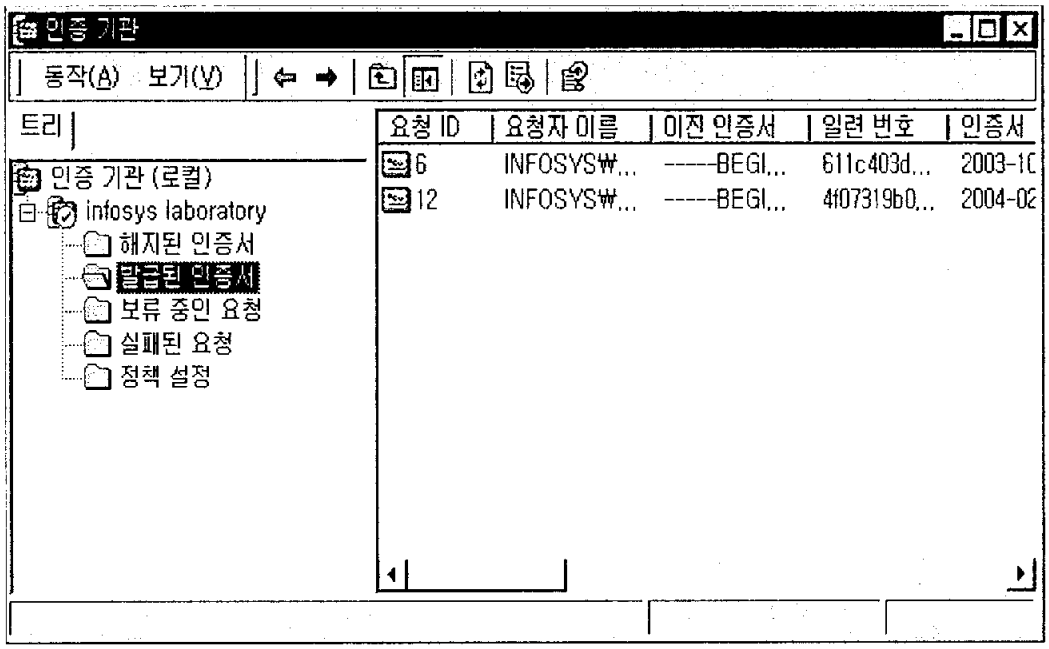
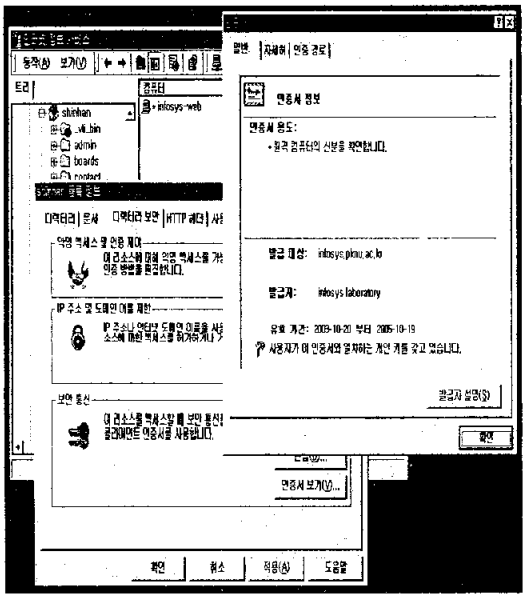


그림 3-15 인증기관에 나타난 발급된 인증서 목록



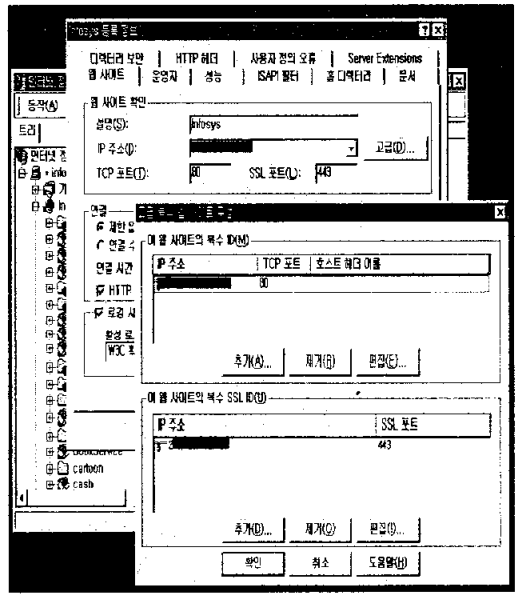


그림 3-16 인증서를 IIS에 설치한 화면 / 그림 3-17 인증서를 설치 후 SSL를 적용시킨 화면

IIS에 Certification Key를 수입하여 설치한 상태를 그림 3-16 에서 보여 주고 있으며, IIS에 Certification Key를 수입 후 SSL을 세팅한 모습을 그림 3-17에서 보여주고 있으며, 이 그림에서 SSL 포트는 443번으로 잡은 모습을 볼 수 있다.

4. SSL을 적용한 전자상거래 시스템 구축

4.1 시스템 개요

OS 는 Microsoft Windows 2000 Server 로 한다. 그리고 웹 서버는 운영체제 내에 있는 IIS (Internet Information Server)를 이용하여 웹 서버와 운영체제 간의 연동을 꾀한다. IIS에 인증서를 수입하여 설치한다. 그리

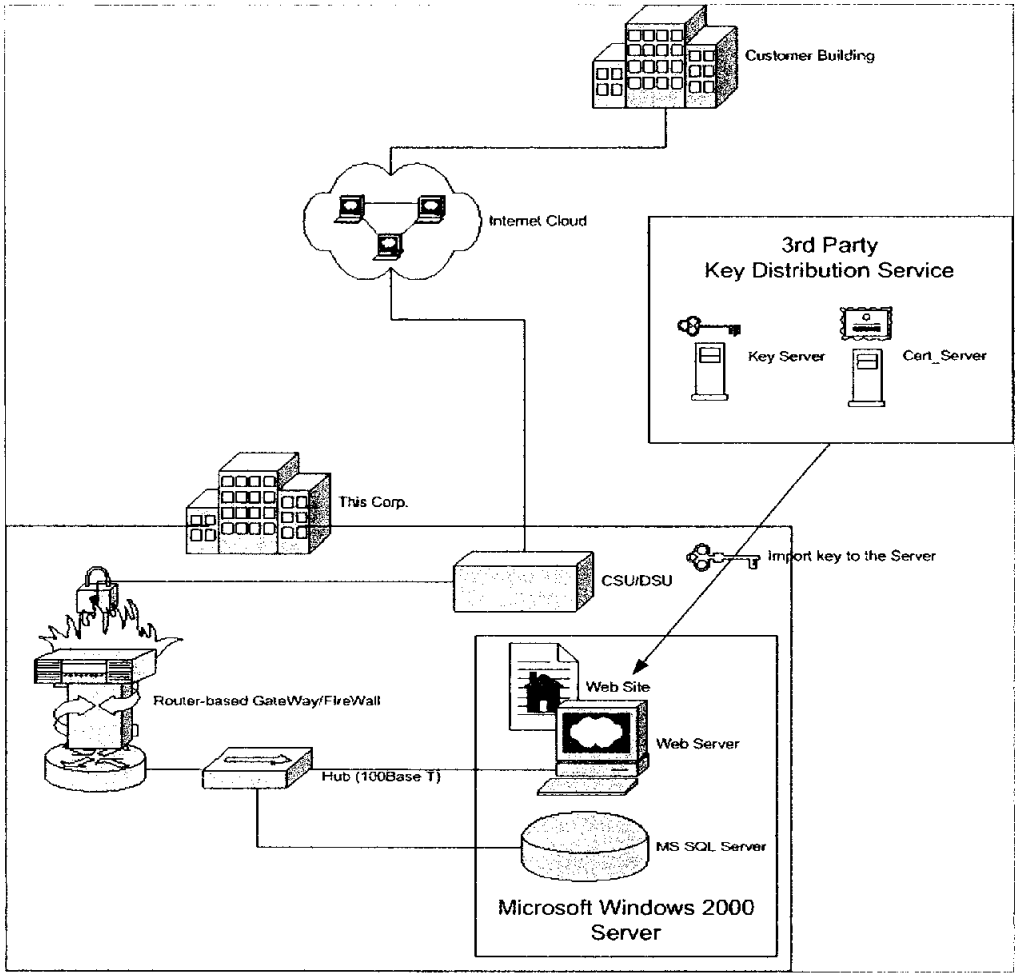


그림 4-1 SSL을 적용한 전자 상거래 시스템의 개념도

고 서버에서 제공하는 인증서 서버 서비스를 동시에 구현한다. 이로서 SSL을 구현하여 클라이언트의 로그인과 더불어 결재시의 보안을 구현하였다.

해당 시스템의 개념도는 그림 4-1에서 나타내고 있다. 개념도에 나와 있는 것처럼 일반적으로는 단독 인증기관으로써 3rd Party 업체를 이용하여 공용 인증서 발급 후 웹 서버에 설치한다. 그러나 여기에서는 Enterprise 인증기관으로 하여, 자체적으로 인증서 Root 인증기관을 설치하고 자체 인증서 발급을 도모하였다.

웹 서버와 데이터베이스 서버는 물리적으로 단일 서버에 위치하게 하였고, 데이터베이스는 회원 정보, 제품 정보를 보존하도록 하였고, 스스로 발급한 인증서를 가지고 직접 웹 서버에 설치하도록 하였다. 웹 인터페이스는 사용자의 주문과 그에 대한 진행 프로세스 출력, 그리고 생산 현장에서의 주문에 대한 offer, order, payment, delivery 정보가 출력될 것이다. 쿠키를 이용한 장바구니 기능을 구현하며 이때의 제품정보들은 앞서 언급한 DB와 연동하여 ASP를 이용하여 출력한다

각 페이지는 자바스크립트를 이용하여 클라이언트 사이드로 제어한다.

4.2 HTTPS 개요

HTTPS는 넷스케이프에 의해 개발되고 넷스케이프 브라우저에 구현된 웹 프로토콜로서, 사용자의 페이지 요청들과 웹 서버에 의해 반환되는 페이지들을 암호화하고 해석한다. HTTPS는 실제로 넷스케이프의 SSL을 정규 HTTP 응용계층 하에서 서브 계층으로서 사용한다. (HTTP가 하부계층인 와 TCP/IP의 상호작용을 위해 80번 포트를 사용하는데 비해, HTTPS는

443번 포트를 사용한다). SSL은 RC4 스트림 암호화 알고리즘을 위해 40 비트 크기의 키를 사용하는데, 이는 상업적 데이터의 교환을 위한 암호화 정도에 적합하다고 간주된다.

HTTPS와 SSL은 서버로부터의 X.509 디지털 증명서 사용을 지원하므로, 필요한 경우 사용자는 발신자를 믿을 수 있음을 증명할 수 있다. SSL은 넷스케이프가 W3C에 표준으로 제안한 개방형이며, 비독점적 프로토콜이다 [22].

4.3 로그인 과정에서 SSL/https 구현

로그인 프로세스에 SSL을 적용하기 위해 앞에서 웹 서버에 인증서를 수입하였고 그 내용을 도시하였다. 일단 웹 서버에 인증서를 수입한 후에는 https를 적용할 수 있도록 해당 프로세스가 수행될 웹 서버의 디렉토리에 SSL을 적용시켰다. SSL 에는 128Bit 암호화 옵션을 그림 4-2에서처럼 걸 수 있는데 이는 낮은 비트의 RSA 암호화를 사용해 보안이 깨질 우려가 있는 민감한 사이트에 적용 할 수 있다.

또한 SSL을 해당 디렉토리에 적용시킨 후 로그인 스크립트의 경로를 https://... 형태의 절대경로로 지정하여 로그인 과정에 SSL이 적용되도록 하였다.

회원 가입, 로그인 부분과 관리자 창 메뉴의 모든 부분을 https 통신으로 그림 4-3과 같이 처리하여 개인 정보 및 관리자 모드에 대한 보안을 처리할 수 있다.

SSL을 이용해 Administrator Mode로 로그인 한 화면은 그림 4-4과 같

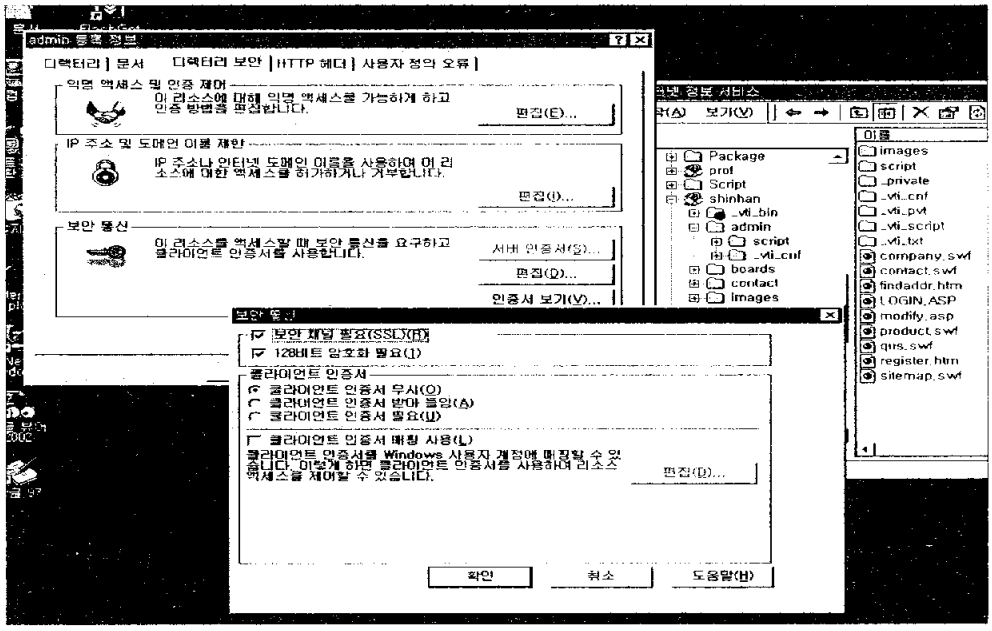


그림 4-2 관리자 디렉토리에 SSL을 적용시킨 화면

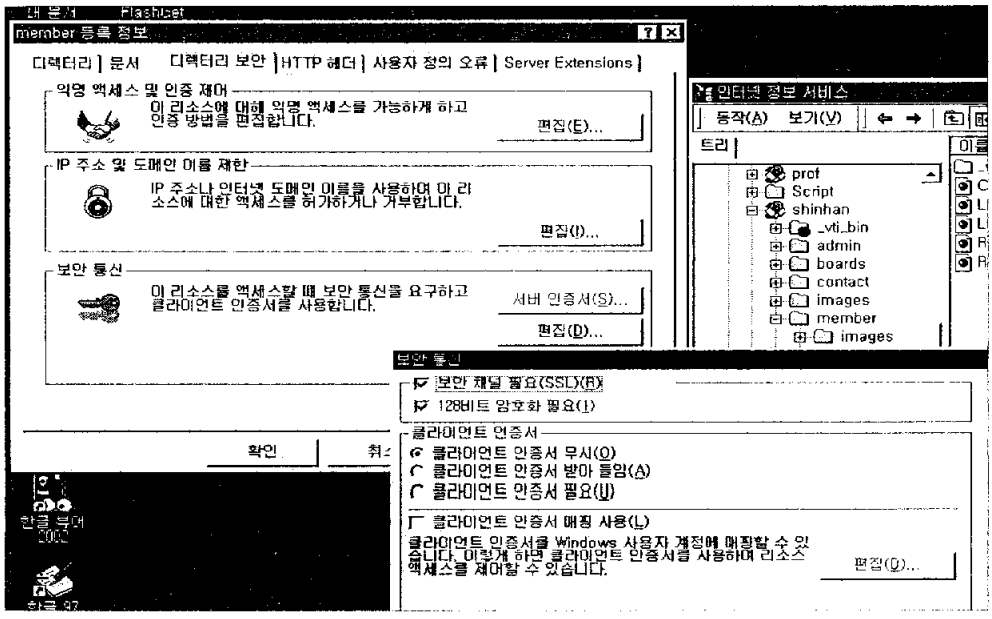


그림 4-3 회원 디렉토리에 SSL을 적용시킨 화면

으며, 화면에 보이는 주소에는 http이지만 로그인 과정에 들어가는 스크립트 및 Administrator Menu 창 전체는 https 프로토콜로 보호되어 있다.

관리자 메뉴는 크게 "회원목록보기", "제품목록보기", "제품추가하기", "주문관리하기" 의 4가지로 구성되어 있으며, 각 메뉴는 별도의 창에서 인터페이스를 제공한다. 기본적으로 관리자 및 기타 로그인시에는 SSL을 적용, https로 통신을 하도록 하여, 정보 노출의 위험을 피하고자 하였다. 이러한 모든 메뉴들은 모두 https 통신을 기반으로 해서 이루어진다.

회원 관리 화면은 그림 4-5과 같이 보여주고 있다. 회원 삭제 기능 및 아이디 클릭시 추가정보화면이 뜨며, mail address를 클릭시 자체 폼메일로 메일을 보낼 수 있다.

제품관리 화면은 그림 4-6과 같이 보여 지는데 제품삭제 기능과 더불어

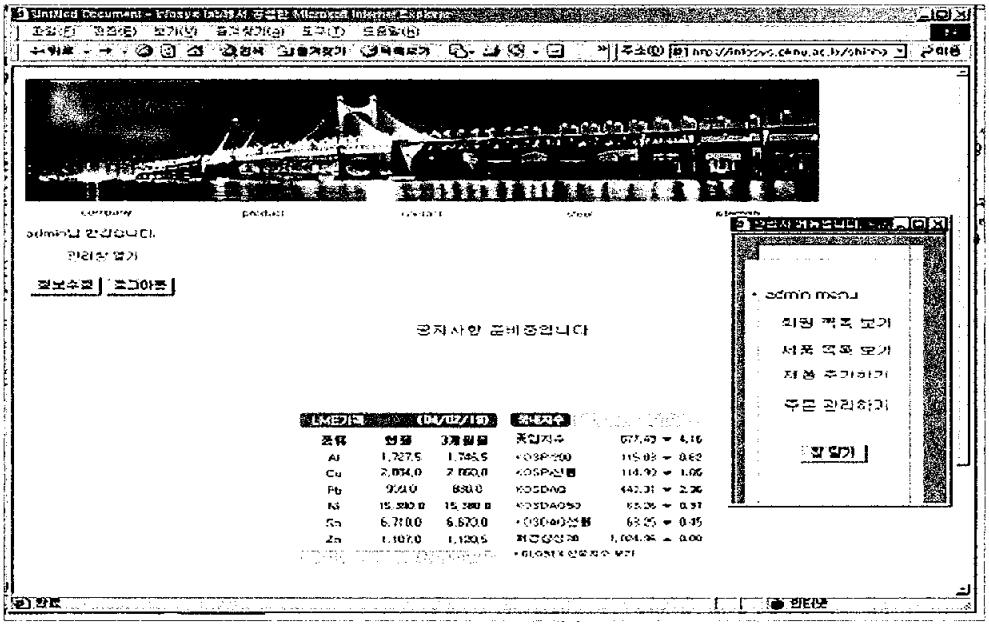


그림 4-4 https 가 적용된 관리자 모드 화면과 관리메뉴 창

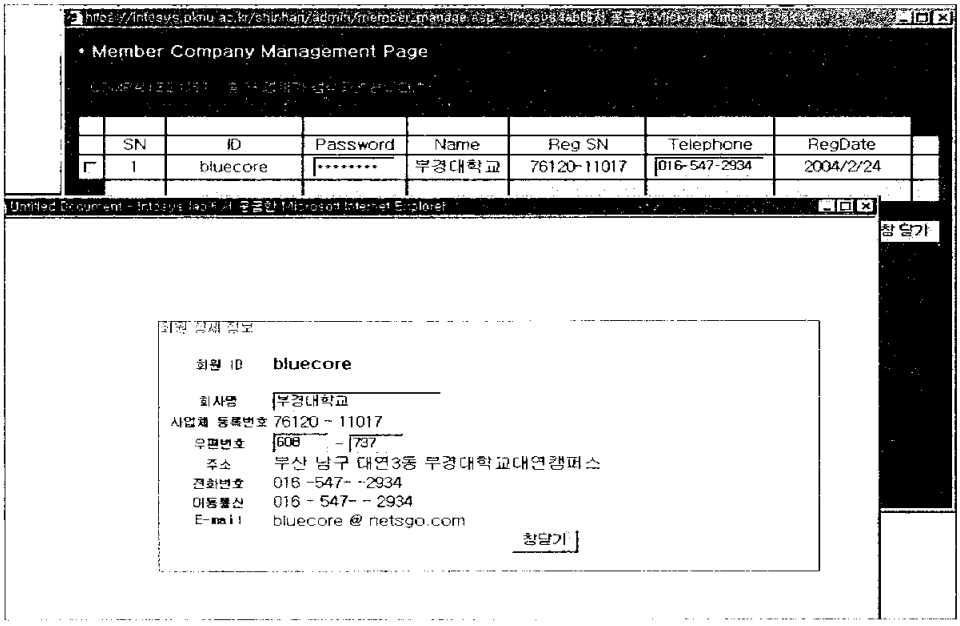


그림 4-5 회원 관리화면

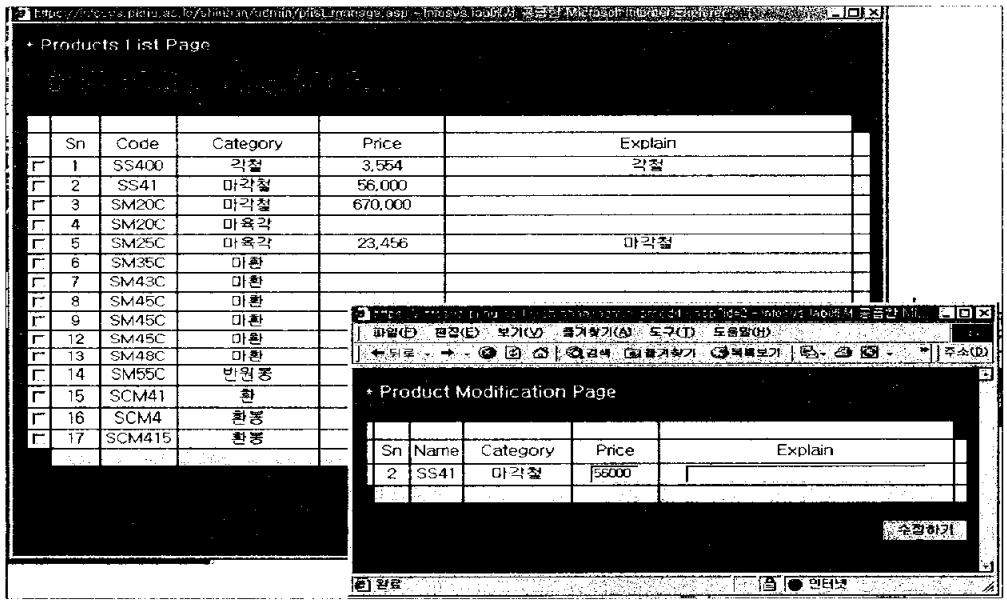


그림 4-6 제품 관리화면

기존제품의 수정도 가능하도록 인터페이스를 제공하고 있다.

제품추가 화면은 그림 4-7과 같이 새로운 제품이 추가되거나 새 카테고리가 추가되었을 시 웹페이지 상으로 추가할 수 있도록 하였다.

주문정보 화면은 그림 4-8과 같이 입금/배송 정보를 추가 저장할 수 있도록 하여, 주문관리를 하도록 하였다.

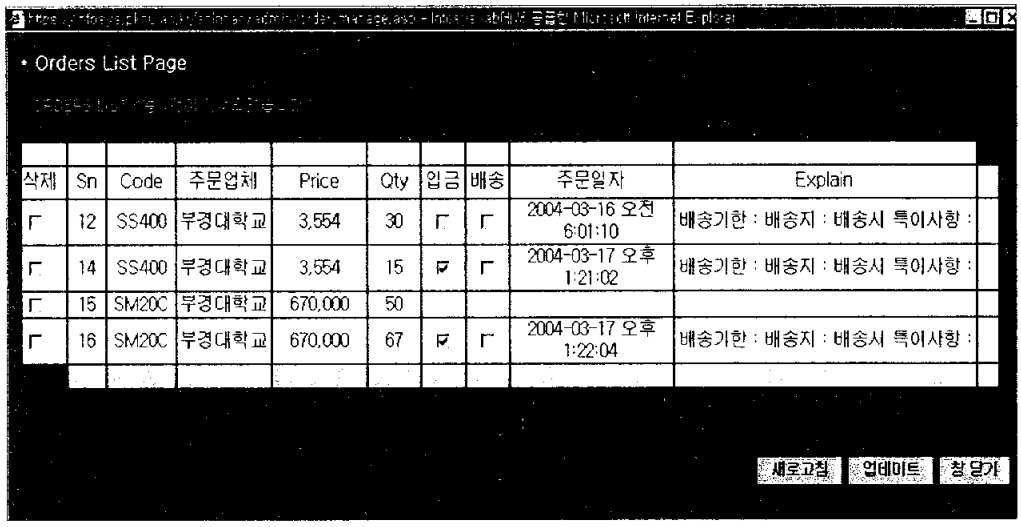


그림 4-7 주문 정보 화면

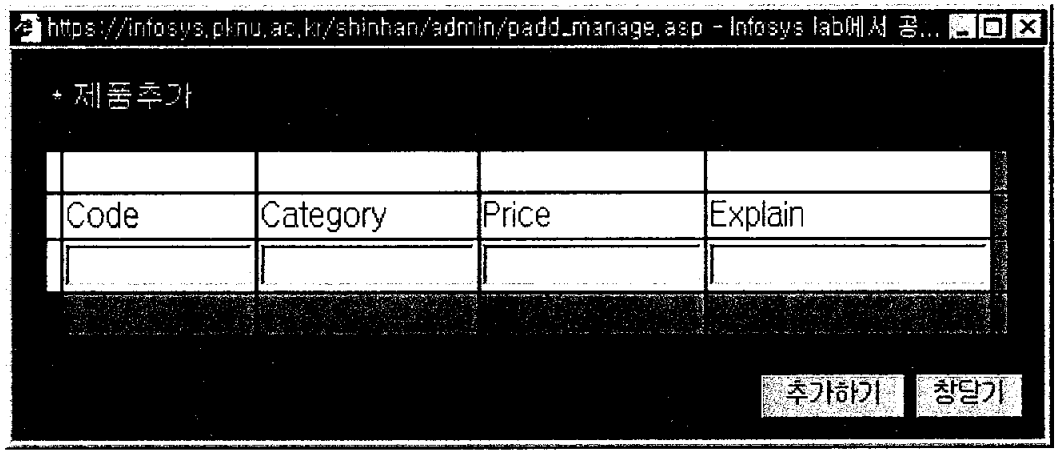


그림 4-8 제품 추가 화면

4.4 상품 거래 과정에서 SSL/https 구현

상품 구매 시 회원의 정보 보안을 위해, SSL을 해당 디렉토리에 적용시켰다. Product 가상 디렉토리에 물품 구매를 위한 정보 보안을 그림 4-9에서와 같이 128bit 암호화하고 SSL을 적용시켰다.

앞에서 언급했던 것처럼 인증서는 하나의 사이트 당 하나의 인증서를 설치할 수 있다. 현재 루트에 인증서가 설치되어 있으므로 가상 디렉토리에는 이 인증서를 적용시킬지에 대한 여부만을 선택할 수 있다. 이 디렉토리에서는 등록정보에서 보안 항목을 SSL 필요, 128비트 암호화 항목을 적용하였다.

비보안 연결로의 리디렉션에 대한 경고문은 그림 4-10과 같이 나타 낼

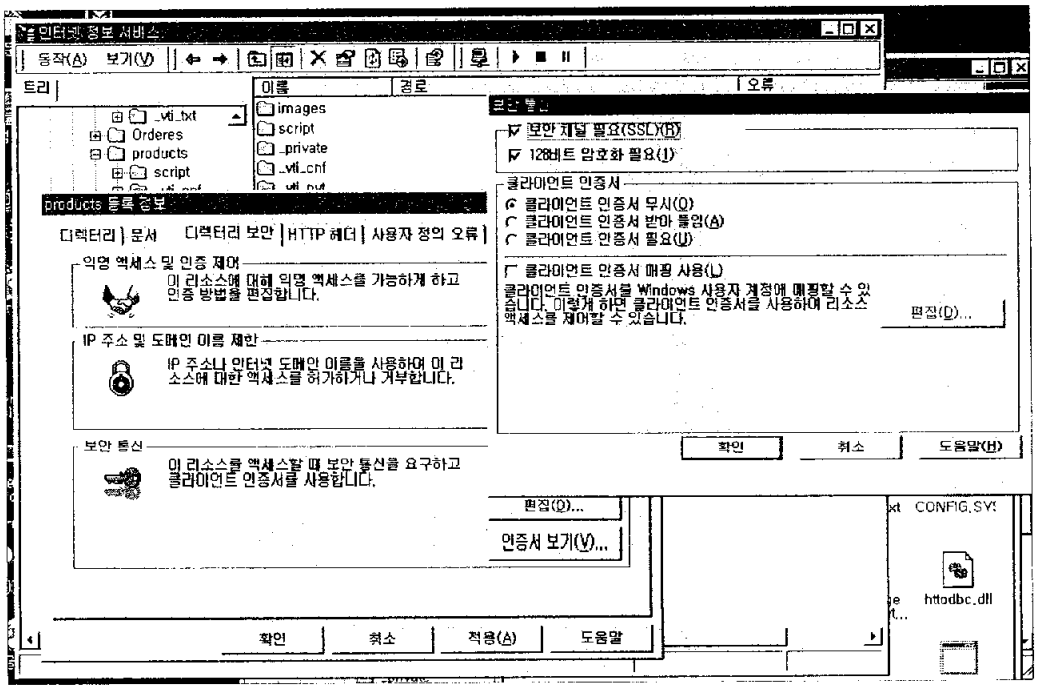


그림 4-9 Product 가상 디렉토리에 SSL을 적용

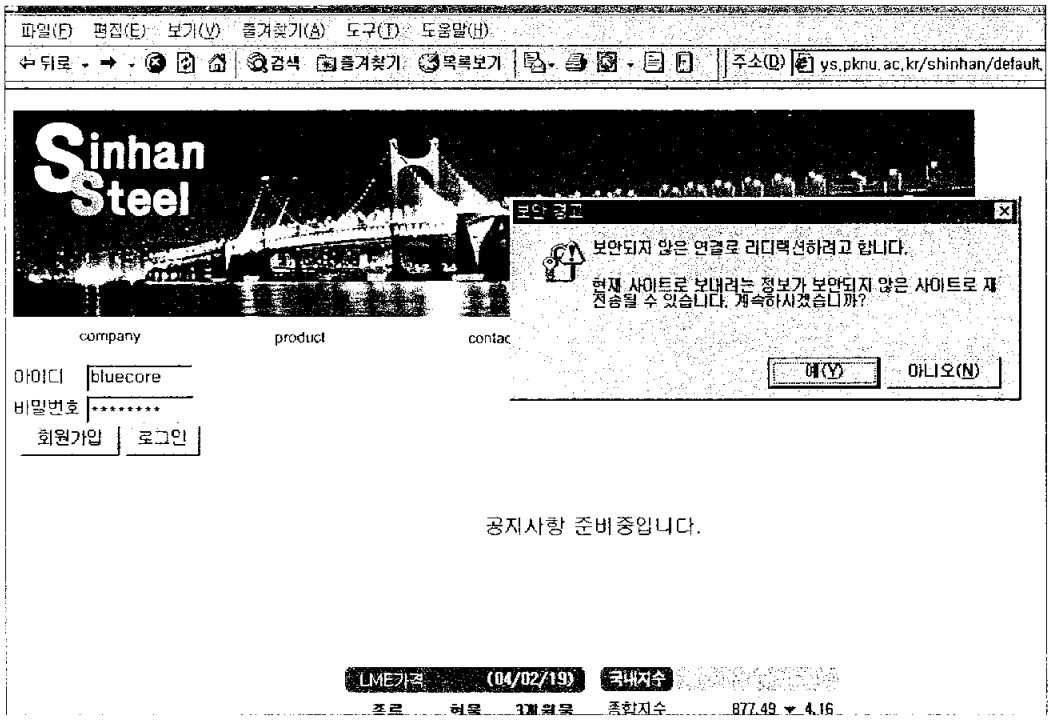


그림 4-10 https를 적용한 로그인 화면

수 있는데, https를 적용하였을 경우 Response.Redirect()(ASP), document.location.href 등을 써서 페이지 전환 시 전환된 페이지가 비 https 페이지면 브라우저는 이러한 경고음을 띄우게 된다.

아래의 항목들은 SSL을 적용한 후의 Product 관련 페이지들이다.

주문은 로그인 후 이용이 가능하도록 하였으며, https를 쓰는 과정에서 웹 프로그래밍상의 문제가 있는데, 페이지를 상대 경로로 결정할 수 없다는 점이다. 이는 웹 서비스에 있어 기본 프로토콜이 http로 정해져 있으므로 상대 경로에 프로토콜을 기재하는 방법이 없기 때문이다. 하지만 일단 같은 웹 디렉토리 내에서는 바로 인식하므로 특별히 문제될 점은 없고, 디렉토리

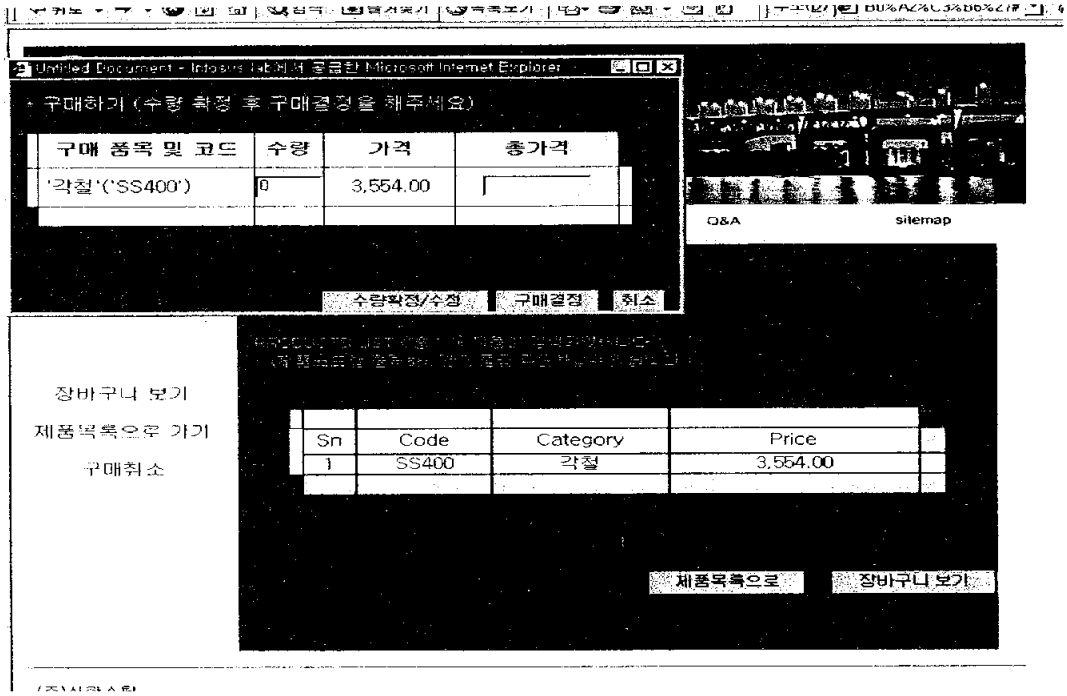


그림 4-11 주문할 물품의 수량과 총 가격을 결정하는 화면

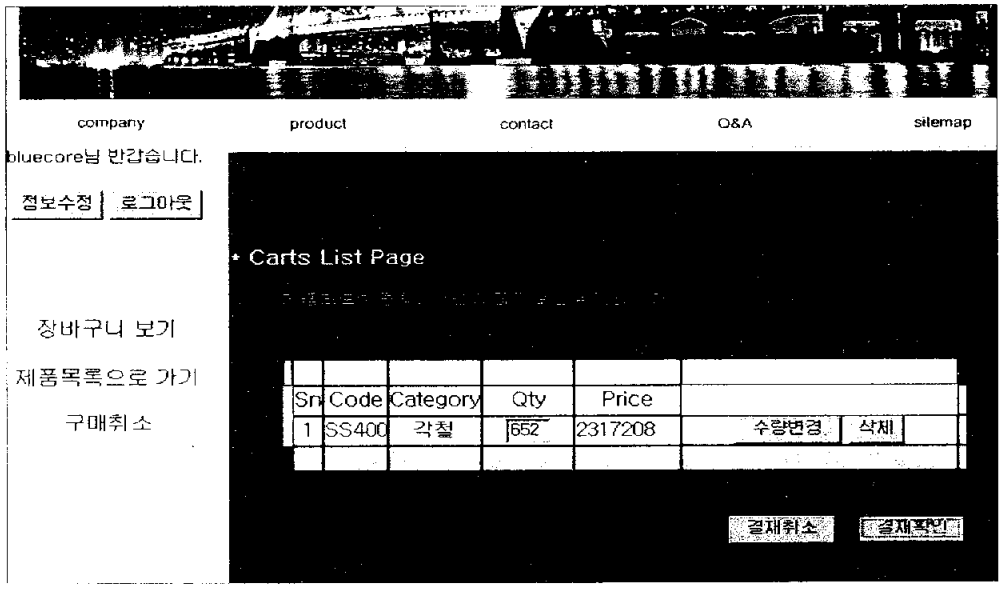


그림 4-12 쿠키를 이용해 장바구니에 기록된 물품의 수량과 총 가격

가 다른 경우에만 주의를 해주면 된다. 구매자가 물품을 선택하여 수량을 결정하면, 총 가격이 산출되는 방식으로 그림 4-11에서 나타내고 있다.

장바구니에 저장된 주문 물품의 각 내역과 총 수량과 총 가격을 나타내는 화면 그림 4-12로, 이 장바구니 폼은 쿠키를 이용해 작성되었으며, 역시 이 과정 전체도 SSL이 적용되어 있는 상태이다.

견적서를 작성하기 위해 필요한 내역을 적는 양식은 그림 4-13에서 나타내고 있다. 신용카드 결제부분은 구현하지 않았다. 하지만 3rd Party 업체와 계약 후에는 바로 여러 가지 형태의 결재방식을 적용할 수 있다. 마지막으로 앞의 과정을 통해 작성된 견적서 내용을 그림 4-14에서 보여주며, 견적서 작성과 동시에 관리자 메뉴에 있던 주문 내역에 동시에 내용이 추가되게 된다.

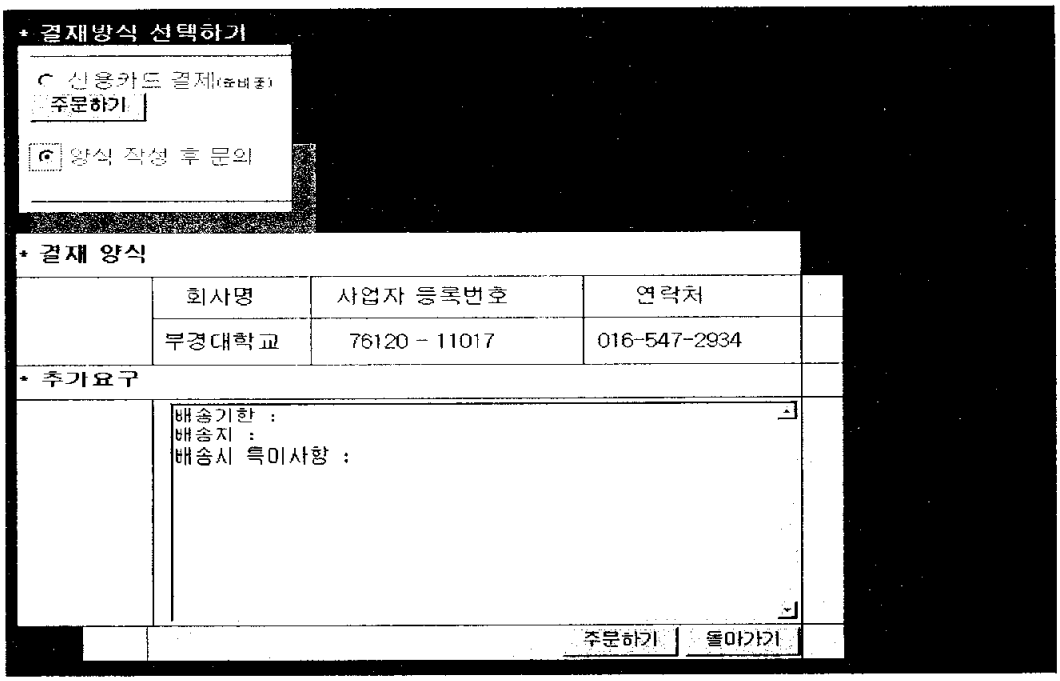


그림 4-13 주문하기위한 결재양식 작성 폼

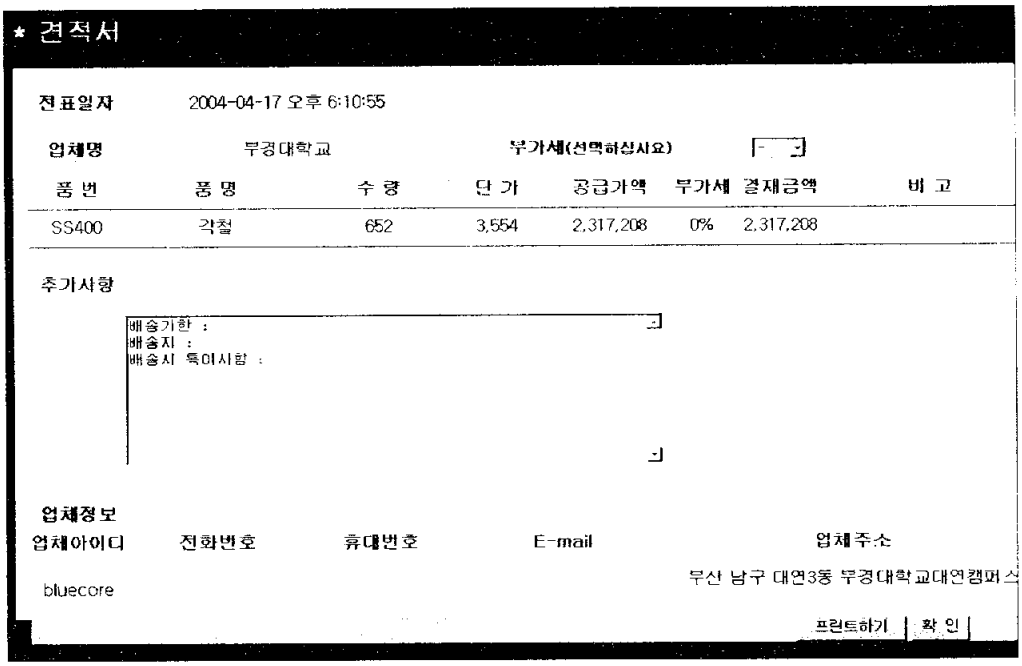


그림 4-14작성된 견적서 내역

지금까지, (주) 신한스틸 전자상거래 시스템이 물품구매, 견적서 처리 등이 잘 작동되며 타 중소기업에서도 널리 활용 가능하다고 판단된다.

5. 결론

전자상거래의 보안은 단순히 시스템을 외부로부터 보호하고, 데이터를 암호화하여 저장하는 것만으로 그치지 않는다. 기업간 전자상거래시스템은 기업과 기업간, 기업과 은행 간의 정보교류를 통하여 이루어진다. 따라서 관련 보안정책은 마켓플레이스에 관련된 참여자는 물론이고 국가 및 세계 전반에서 효용성을 가져야 한다. 또한 개발되는 보안 기술과 표준들은 하나의 응용이 아니라 B2B 플랫폼으로서의 작용을 할 수 있어야 한다.

이러한 관점에서 본 논문에서는 전자상거래 시스템용 통합 인증시스템을 구축하기 위한 알고리즘 선택법과 SSL을 적용하기 위한 구축 방안을 제시하였다. 먼저 인증 시스템을 구축하기 위한 모델링을 고찰하였다. 인증서버의 형태는 적은 수의 서버로 시스템을 운영할 수 있는 엔터프라이즈형 독립 인증서버 가 가장 적합함을 알 수 있었다. 키 생성알고리즘으로는 가장 보편적으로 쓰이면서도 강력한 암호화를 제공하는 RSA 알고리즘이 적합하고, 키를 분배하는 방식은 비대칭형 방식이 효과적임을 확인하였다.

또한 SSL이 어떠한 위치에서 작동하는지를 확인하고, 여기에 인증서 프로세스를 구현하고 RSA 알고리즘을 기반으로 하는 SSL 서비스 시스템을 구현하였다.

마지막으로 제안한 내용의 유용성을 확인하기 위하여 Microsoft Windows 2000 서버를 기반으로 SSL을 적용하여 중소기업용 통합 인증서버를 구축하였다. 또한 (주)신한스틸 전자상거래 사이트를 구성하고 로

그인 부분과 상품주문 및 결재, 견적서 작성 등을 실제 운용하여 인증시스템과 SSL의 적용 결과를 검증하였다.

그 결과 자체 인증서 발급 및 갱신이 가능하고, 인증서버 및 웹 서버가 OS에 통합 운영되어 효율적이고 비용절감이 가능함을 알 수 있었다. 본 시스템은 중소기업의 환경에 맞춘 전자상거래시스템으로서 중소기업의 경쟁력 재고에 의미가 있고, 차후 다른 형식의 전자상거래시스템의 개발에 있어서 보안기법의 적용을 통해 다양한 방법으로 활용될 수 있을 것으로 생각된다.

향후의 연구과제로는 본 연구에서 구현한 윈도우 서버 플랫폼뿐만이 아니라 기타 플랫폼에 적용시키는 방안에 대한 연구가 있어야 할 것이다. 또한 지금의 RSA 알고리즘은 강력하지만 방대한 양의 연산을 함으로써 서버의 리소스를 잠식하고, 성능을 저하시키는 문제가 남아있다. 이를 해결하기 위한 보다 가벼운 알고리즘이나 독립 연산 서버를 통해 서버의 부담을 줄이는 방법에 대한 연구가 필요하다고 생각한다.

6. 참고문헌

[1] 김철환, "CALS/EC 와 정보화 사회," 한국전자거래학회지 Vol.1 No.2 pp.1-4, 1996

[2] 유완재, "CALS/EC 추진 현황과 과제," 한국전자거래학회지 Vol.1 No.2 pp.1-3, 1996

[3] 김성희, 장기진, "IDEF0 와 개념적 설계를 이용한 물류 CALS/EC 구축 방안 연구," 한국전자거래학회지 Vol.4 No.1 pp.37-59, 1999

[4] 이봉환,윤동원,김칠민,채융웅,김현곤, "정보보안: 카오스 암호화 알고리즘을 이용한 웹 보안 시스템 설계 및 구현," 한국정보처리학회 논문지 Vol.30 No.2-1 pp.883-885, 2003

[5] 강영구,류성열, "전자상거래 보안을 위한 Y2K 암호 알고리즘의 설계," 한국정보처리학회 논문집 Vol.7 No.10 pp.3138~3147, 2000

[6] 정은희, 이병관, "ECSSL 기반 DIT 에이전트," 한국정보처리학회 논문지 Vol.B9 No.5 pp.599, 2002

[7] 임선간,권태경,윤재호,천동현, "IPSec,VPNs vs. SSL VPNs," 한국정보보호학회지 Vol.13 No.5 pp24-31, 2003

[8] 최철림외 2명, "웹 기반 사용자 정보 암호화 알고리즘 설계 및 구현," 한국정보처리학회 Vol.30 No.2-1 pp733-735, 2002

[9] 김학배, 남의석, 민병조, "인터넷 침해 대응기술 : 리눅스 커널 쓰레드에 기반한 SSL 웹 암호화 가속기 개발," 정보처리학회 논문지C Vol.10 No.5 pp 657-660, 2003

[10] 조은애등 4명, "SSL 컴포넌트의 설계 및 구현," 정보처리학회지 Vol.30 No.1-2 pp883-885, 2003

[11] 박영민, 강민섭, "SSL을 이용한 암호 및 인증서버/클라이언트의 구현," Vol.30 No.2-1 pp841-843, 2003

[12] 이은석, 이진구, "다중 에이전트 기반의 지능형 전자상거래 시스템," 한국정보처리학회논문지C Vol.8-C No.6 pp.855-864, 2001

13) (주)퓨쳐시스템 "암호체계센터 암호 알고리즘 및 프로토콜의 이해," http://cnscenter.future.co.kr/crypto/algorithm/intro/intro.html#intro

14) Microsoft사, "Microsoft Windows 2000 Online 도움말," 항목 : SSL을 이용한 서버인증

15) 박영호, "ECDSA를 적용한 ID 기반의 사용자 인증 및 키 교환 프로토콜," 정보보호학회논문지, Vol.12 No.1 pp3-6, 2002년

16) 이성재등 3명, "SEED 구현 적합성 검증 시스템에 관한 연구," 정보보호학회 논문지, Vol.13 No.3 pp.69-86, 2003년

17) Microsoft사, "Microsoft Windows 2000 Online 도움말," 항목 : RSA / 암호화 알고리즘

18), 19) Microsoft사, "Windows2000 공개키 비공개 키 기반 구조 소개 백서," 1999

20) 김종상, 전화숙 공역, "데이타통신 및 컴퓨터통신," 제2장 pp.50-61 사이텍미디어, 2001

21) 정은희, 이병관, "ECSSL 기반 DIT 에이젠트," 한국정보처리학회 논문지, Vol.9-B No.5 pp.599-607, 2002

22) 텀즈(www.terms.co.kr) : 항목-SSL http://terms.co.kr/SSL.htm

처음 이곳을 발 딛었을 때 모든 것이 낯설고 어색하였는데
그 어색함으로 시작한 대학원 생활이 이제는 막바지로 향해가고
있습니다.늦은 나이에 새로운 배움의 시작이 회사생활과 병행하여
이루어낸 결실이기에 제 자신에게는 그 기쁨이 배가 되는가
생각됩니다.

먼저 미흡하나마 본 논문이 완성되기까지 많은 가르침과 마무리
까지 이끌어준 김석태 지도교수님께 머리 숙여 감사드리며, 심사위원
으로 지도를 아끼지 않은 고 장주석 교수님과 마무리까지 도와주신
박규철 교수님께 감사를 드리며, 또한 전공과 전혀 무관한 새로운
분야를 체계적으로 지도해 주셨던 김성운 교수님, 하덕호 교수님,
정연호 교수님 정신일 교수님, 윤종락 교수님께도 감사를 드립니다.

논문을 준비하는 동안 진심으로 도와준 이수명, 문종현등 정보시스
템 연구실 가족들에게도 감사드립니다. 직장생활 가운데서 학업을
마칠 수 있도록 물심양면으로 도와주신 강필대 국장님, 강호경 부장
님께 진심으로 감사를 드리며, 학교 때문에 회사생활에 충실하지 못
했던 부분을 말없이 도와주신 토목부 직원과 망건설국 직원들에게 다
시 한번 감사를 드립니다.

마지막으로 마음으로 적극적으로 도와주신 어머님과 항상 옆에서
힘이 되어준 아내와 공부를 핑계로 소홀히 했던 명재, 명진이 그리고
나를 사랑해주시는 모든 분들에게 감사를 드립니다.

이제 배움의 마무리 시점에서, 빠르게 변화하는 정보통신분야에서
이 배움의 마무리가 끝을 의미하는 것이 아니라 또 하나의 새로운 출
발점이 될 수 있도록 다짐합니다.