이중 방법을 지원하는 임베디드 보안 팩스 서버 개발

이 상 학[†] · 정 태 충[††]

요 약

지난 수십 년간 문서 전송 수단으로 사용되어 왔던 팩스는 인터넷의 출현으로 전자우편, 파일 전송 규약 등이 널리 이용되는 현재에도 여전히 중요한 데이터 통신 수단이다. 인터넷과 마찬가지로 팩스 데이터의 보안에 대한 필요성은 높지만 그 연구는 인터넷에 비해 미비한 상태이다. 본 논문에서는 일반 팩스의 사용을 그대로 유지하면서 보안성을 높일 수 있는 임베디드 보안 팩스 서버의 하드웨어, 소프트웨어의 개발에 대해 기술한다. 개발된 시스템은 데이터의 암·복호화에 대한 지연을 최소화 하면서 보안을 제공할 수 있도록 설계되고 구현되었다. 암호 알고리즘은 국제 규격과 자국의 규격이 있으며, 각 국가마다 정책적 제한을 두고 있기 때문에 국내에서 인정되는 표준 암호 알고리즘을 적용하였다. 또한 보안(Security) 모드와 비보안(Non-Security) 모드의 이중 동작 방법을 지원하여 사용자가 선택적으로 이용할 수 있도록 하였다. 정부, 기업 등의 실사용자의 기술적 요구사항을 반영하여 실제 직접 적용될 수 있는 시스템을 개발하였으며, 실제 환경 하에서 시험을 거쳐 성능 및 기능에 대해 검증하였다.

Development of Embedded Security Fax Server Supporting Dual Mode

SangHak Lee[†] · TaeChoong Chung[††]

ABSTRACT

Even though the Internet applications such as e-mail and FTP are widely used, fax is still an important media for data communications till today. Many researches on security over the Internet data communications have been done over the years, on the other hand not many researches have been dedicated to the fax security issue which is as important as the Internet. In this paper, we describe the development of hardware and software of the embedded security fax server which increases the security in supporting existing fax. The developed system is designed and implemented to maintain security while minimizing the delay due to encryption · decryption. Since there's international or domestic cryptographic standard and each nation have their policy to restrict the use of cryptographic system, we adopt domestic standard cryptographic protocol admitted in Korea. And the system supports two modes : Security mode and Non-Security mode that user can choose from. The system can be applied directly which is the requirements of users at company and the government. We verify the performance and functioning of the system in various real environment.

키워드 : 암호 통신(Cryptographic Communication), 팩스 서버(Fax Server), 보안(Security), 임베디드 시스템(Embedded System), 이중 방법(Dual Mode)

1. 서 론

인터넷의 확산은 일반 사용자의 문서 전송 방식에도 많은 변화를 가져왔다. 인터넷 사용 이전에는 일반 사무실이나 가정에서 문서를 주고받는 가장 보편적인 수단이 팩스였으나 인터넷의 출현으로 e-mail, FTP, WWW 등의 애플리케이션을 사용하여 문서 파일을 전송하기 시작했다. 그러나 아직까지 오피스에서는 여전히 팩스를 구비하고 있으며 종이로 된 문서를 전송하는 일반적인 수단이다[4]. 우리나라는 초고속인터넷 보급률이 높은 나라 중의 하나이기 때문에 인터넷에 쉽게 접속할 수 있지만 대부분의 국가들은 그렇지 못하다. 이러한 국가들과의 문서 전송을 위해서는 전화망의 팩스를 사용해야만 한다. 이와 같이 아직까지 문서 전송의 대표적인 수단인 팩스를 사용할 때, 인터넷과 비교해 문제가 되는 부분은 기밀을 요하는 문서를 보낼 수 있는 보안방법이 부재하다는 것이다. 인터넷상에서는 다양한 보안을 위한 암호 알고리즘들이 개발되어 사용자가 쉽게 이용할 수 있다. 그러나 보안을 위해 암호화 기능이 탑재된 팩스는 쉽게 구할 수 없으며, 이로 인해 팩스를 통한 문서 전송 시 보안이 된 문서를 전송한다는 것은 불가능에 가깝다.

현재 전 세계의 기업들은 정보전쟁을 하고 있다고 해도 과언이 아니다. 기업의 핵심 정보들을 지키려는 노력뿐만

† 정 회 원 : 전자부품연구원 유비쿼터스컴퓨팅연구센터 선임연구원, 경희대학교 대학원 컴퓨터공학과 박사과정
†† 정 회 원 : 경희대학교 컴퓨터공학과 정교수
논문접수 : 2004년 4월 22일, 심사완료 : 2004년 6월 8일

아니라 경쟁기업의 정보를 얻기 위한 시도를 하고 있다. 자칫 직원들의 작은 부주의 혹은 고의적 정보 유출은 기업에게 치명적인 타격을 줄 수 있다.

본 논문에서는 기업의 중요한 문서 전송 수단이지만 그 보안성은 매우 취약한 팩스의 약점을 보완하기 위해 개발된 보안용 팩스 서버에 대해 기술한다. 본 논문에서 개발된 팩스 서버는 기존에 사용하고 있는 환경에 영향을 미치지 않으면서 쉽게 보안기능을 추가할 수 있는 형태이다. 즉, 기존의 사용자가 사용하는 팩스와 이 팩스가 연결되어 있는 전화망 사이에 놓여 암복호화를 수행하는 자립형(stand-alone) 임베디드 기기이다. 이는 기존의 팩스를 그대로 사용하며 보안성을 확보되기를 원하는 요구사항을 만족하는 것이다. 송신단과 수신단에 짝을 이루어 사용되게 되며 보안 모드와 비보안 모드의 두 가지 모드를 지원한다.

연구된 부분은 사용자가 이용하기 쉬우면서도 보안성은 매우 뛰어난 팩스 서버가 갖추어야할 조건을 설정하여 이 기준에 맞는 시스템을 디자인하고 구현하였다. 본 논문의 구성은 다음과 같다. 2장에서는 보안 팩스를 위한 관련연구에 대해 알아보고, 3장에서는 본 논문에서 개발된 시스템의 하드웨어, 소프트웨어 구조 및 이의 구현에 대해 기술하였으며, 4장에서는 개발된 시스템의 시험 결과를 기술하고, 마지막으로 5장에서는 결론 및 향후 연구되어야할 내용을 알아보았다.

2. 관련 연구

본 장에서는 보안 팩스와 관련된 연구들에 대해 기술한다. 우선 기존의 팩스 보안을 위한 표준화 및 관련 시스템에 대해 알아보고 보안을 위한 암호화 기술에 대해 기술한다. 팩스 데이터의 암호화를 위해 사용될 수 있는 암호 알고리즘과 이를 자립형 시스템(stand-alone system)으로 구축하기 위한 임베디드 기술에 대해 알아본다.

2.1 팩스의 보안기술

팩스 송신자와 수신자간의 통신과 관련된 보안문제는 크게 두 가지이다. 첫째는 송신자나 수신자가 아닌 제3자가 팩스 데이터를 가로채거나 변조하는 것이고, 둘째는 인증문제이다. 즉 송신자가 문서를 보내는 당사자인가 하는 것과 수신자는 송신자가 문서를 받기 원했던 그 본인이냐 하는 것이다[1]. 이와 같은 문제를 해결하기 위해 일반적으로 암호시스템을 사용한다. 즉, 암호키를 이용하여 원본 데이터를 뒤섞어 알아 볼 수 없는 암호문으로 만든 후 전송하고 수신측에서는 이와 같은 암호화된 문서를 암호키를 이용하여 원본 데이터로 복구한다. 이에 대한 좀 더 기술적인 내용은 다음 절에서 다루기로 한다.

보안 팩스 시스템은 machine-to-machine 간의 보안과 person-to-person 간의 보안으로 구분할 수 있다[3]. 일반적으로 동일한 팩스를 여러 명이 공유하여 사용하기 때문에 같은 팩스에 수신된 문서라 할지라도 수신자 자신을 증명한 후에 문서를 수신할 수 있다. 이러한 팩스 시스템은 수신된 암호화된 문서를 저장할 수 있는 메모리와 스마트 카드 기능을 구비하고 있다.

보안 팩스에 대한 표준화 작업은 IUT-T의 Study Group 8에서 수행되어 Group 3 팩스 표준안을 위한 두 가지 안이 제안되었다. 첫째는 프랑스에서 제안된 RSA 알고리즘 기반 암호시스템이다. RAS는 Ron Rivest, Adi Shamir, Leonard Adleman에 의해 고안된 공개키 암호 알고리즘이다. RSA는 이들의 이름의 앞 글자를 따서 만들었다. 둘째는 영국에서 제안된 HKM/HFX 암호시스템이다. HKM/HFX 시스템은 Hawthorne Key Management(HKM) 시스템, Hawthorne Fascimile Cipher(HFX40), HFX40-I message integrity 시스템을 가리킨다. 두 가지 보안 팩스에 대한 제안은 팩스 프로토콜에 포함되어 Group 3 보안 팩스 권고안이 되었다. 이렇게 해서 나온 Annex G와 Annex H를 포함하는 권고안 T.30과 새로운 권고안 T.36이 ITU-T에 의해 1997년 7월에 승인 받았다[7].

보안 팩스의 가능한 구현은 다음의 몇 가지 범주로 구분할 수 있다.

- 암호 기능이 내장된 자립형(stand-alone) 팩스
- 팩스와 전화망 사이에 연결되는 암호 모듈
- PC에서 동작하는 암호화 팩스 애플리케이션. 이 경우 암호화는 소프트웨어로 수행된다[5].
- 팩스 혹은 팩스 모뎀 소프트웨어를 위한 암호화 툴키트
- 암호 응용 프로그램 인터페이스(APIs)

현재 시중에 나와 있는 팩스 통신의 보안을 위한 기기는 거의 전무하며 구입 가능한 제품들도 대부분 표준을 따르지 않고 독자 기술로 구현되었다. 본 논문에서 구현한 시스템은 위 암호 시스템들 중 두 번째 방식인 팩스와 전화망 사이에 놓여 암복호화를 수행하는 자립형(stand-alone) 임베디드 보안 모듈이며 일반적인 팩스 전송과 보안 전송을 지원한다[6].

팩스의 또 다른 발전방향은 인터넷을 통한 팩스의 송수신에 대한 접근방법이다. 이는 인터넷의 보급이 보편화되었다고 하지만 팩스가 여전히 널리 쓰이고 있기 때문에 일반 전화 교환망이 아닌 인터넷을 이용하거나 팩스가 없을 때 이를 PC에서 에뮬레이션하려는 시도이다. 인터넷 상에서의 팩스 전송을 위해 인터넷 엔지니어링 태스크 포스(Internet Engineering Task Force : IETF)에서 규약 RFC2542를 제정하였다[9]. 본 규약에는 인터넷을 기반으로 하는 다양

한 형태의 팩스 전송에 대해 정의하고 있다. 이외에 팩스를 가지고 있지 않은 인터넷 사용자를 대상으로 팩스 송수신을 애플리케이션에서 가능하도록 하는 서비스[10, 11]나 PC 상에 팩스 모뎀을 장착하여 팩스로 사용할 수 있는 프로그램들이 존재한다[12]. 그러나 팩스 송수신의 종점(end point)이 인터넷에 바로 접속되어 있다면 인터넷의 보안 수준을 보장받을 수 있지만 인터넷을 거쳐 다시 일반 전화 교환망의 팩스로 전송해야 한다면 이 마지막 경로의 보안은 이루어지기 어렵다. 따라서 본 연구에서 개발된 보안 팩스 서버는 이러한 새로운 형태의 팩스 전송에 대해서도 보안성을 보완할 수 있다. 향후 인터넷 팩스의 사용이 점차 증대되리라 예상되지만 스캐너, 프린터, 통신 모뎀이 단일 시스템 내에 존재하는 현재의 팩스는 사용의 편리성으로 인해 여전히 문서 전송의 주요 수단으로 이용될 것이며 이에 대한 손쉬운 보안 방법을 필요로 한다.

2.2 암호 시스템

암호 시스템은 크게 두 가지로 구분할 수 있다. 대칭키 암호시스템과 공개키 암호시스템이 그것이다. 1970년대 이전에 일반적인 암호시스템은 단일키 암호시스템이었다. 대칭키 암호시스템은 암호화키와 복호화키가 같은 하나의 키이다. 이러한 암호시스템을 비밀키 암호시스템 혹은 대칭키 암호시스템이라 한다. 단일 키 암호시스템은 평문을 암호문으로 만들 수 있는 공통의 단일키 혹은 비밀키를 가진 자만이 암호문을 평문으로 복호화할 수 있기 때문에 인증 문제가 자연스럽게 해결된다. 문제점은 송신자와 수신자간의 비밀 키의 안정된 교환방법이다. 이는 암호화된 메시지가 비보안 채널을 통해 전송되기 전에 보안 채널을 통해 키가 전송되어야 함을 나타낸다. 키가 안전하게 교환될 수 없다면, 그 이후의 키를 사용한 전송문은 도청되어 복호화될 수 있다. 키 생성, 전송, 저장을 키 관리라 부르며 이는 모든 암호시스템에 공통된다.

공개키(혹은 비대칭) 암호시스템은 키 관리 문제를 해결하기 위해 1976년 Whitfield Diffie와 Martin Hellman에 의해 고안되었다. 공개키 시스템에서는 모든 사용자가 두 개로 이루어진 한 쌍의 키를 가지는데, 하나는 공개키이고 하나는 개인키이다. 공개키는 공표되고 개인키는 안전하게 보관한다. 모든 통신은 수신자의 공개키를 이용하여 암호화된 후에 전송되기 때문에 송신자와 수신자가 하나의 비밀 키를 공유할 필요가 없다. 개인키의 분배가 필요 없기 때문에 교환되는 키에 대한 가로채기는 문제가 되지 않는다. 대표적인 공개 키 기반 암호 알고리즘에는 RSA 알고리즘이 있다[8].

암호 시스템은 국가에서 정책적으로 관리하고 있으며 인가를 받아야 사용 가능하다. 따라서 본 연구에서는 국내의 사용 목적을 위해 128비트 블록암호알고리즘인 SEED를 사용하였다.

2.3 임베디드 시스템

임베디드 시스템은 미리 정해진 특정 기능을 수행하기 위해 컴퓨터의 하드웨어와 소프트웨어가 조합된 전자 제어 시스템을 말하며 필요에 따라서는 일부 기계(mechanical parts)가 포함될 수 있다[2]. 현재 널리 사용하고 있는 PC는 매우 강력한 컴퓨팅 능력을 지니고 있어 다양한 응용 애플리케이션이 수행될 수 있는 범용성을 지니고 있다. 이에 비해 임베디드 시스템은 전용 기능만을 수행하기 위해 최적화된 하드웨어, 소프트웨어로 구성된다. 따라서 전체 시스템 가격이나 전력을 낮추기 위해 시스템에 많은 하드웨어적인 제약을 가하고 범용 운영체계를 사용하기 보다는 일반적으로 특화된 실시간 운영체계를 사용한다.

이러한 임베디드 시스템의 대표적인 응용 분야들로는 정보가전, 사무기기, 공장자동화, 가정자동화 등의 제어, 이동전화, PDA, 스마트폰 등의 모바일 디바이스 등이 있다.

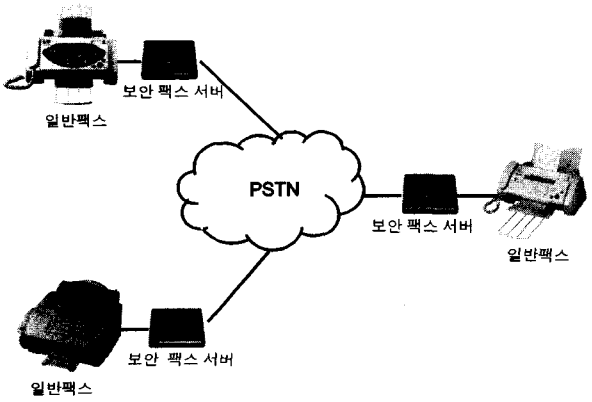
임베디드 시스템을 구성하는 하드웨어는 마이크로프로세서, 메모리, DSP, LED 등이 있다. 보통 마이크로프로세서의 크기나 성능에 관계없이 마이크로프로세서가 삽입된 시스템은 임베디드 시스템이라 하지만 일반적으로 32비트 이하의 마이크로프로세서를 사용한 시스템을 일컫는다. 그 외, 시스템 소프트웨어의 저장, 동작을 위한 메모리가 포함되며, 그 동작 업무에 따라서 특정 DSP가 들어가기도 한다. 사용자와의 인터페이스는 복잡한 구현보다는 단순한 LED 몇 개와 액정디스플레이로 상태를 표시한다.

임베디드 시스템에 들어가는 운영체계는 초기에는 시스템이 단순하여 필요하지 않았지만, 시스템 자체가 커지고 네트워크나 멀티미디어가 시스템에 장착되면서 기능이 복잡해졌기 때문에 실시간 운영체계가 도입되었다. 상용 RTOS(Real-Time Operating System)으로는 윈드리버사의 VxWorks, 마이크로텍의 VRTX, 마이크로웨어의 OS-9, QSSL의 QNX 등이 많이 사용된다. RTOS는 아니지만 임베디드 시스템의 운영체계로 사용되는 또 다른 예는 마이크로소프트의 WinCE와 임베디드 리눅스가 있다. 임베디드 리눅스는 상용 운영체계보다는 실시간적 요소를 만족시키지는 못하지만 오픈소스에 라이선스 비용이 없다는 장점 때문에 그 사용 비중이 점점 커지고 있다. 본 논문의 시스템은 운영체계로 임베디드 리눅스를 기반으로 하여 개발되었다.

3. 시스템 설계 및 구현

본 시스템은 일반 팩스와 전화망 사이에 놓여 팩스 데이터가 전화망을 통해 전달되기 전에 암호화를 수행하여 암

호화된 데이터를 모뎀 통신을 통해 수신측의 보안 팩스 서버로 송신하면, 수신측의 팩스 서버는 이를 받아 복호화를 수행한 후 일반 팩스 데이터를 수신측 팩스로 보내게 된다. 이와 같은 기능을 충실히 수행하기 위해 시스템을 설계 및 구현하였다. (그림 1)은 시스템의 동작 환경도를 나타낸다.



(그림 1) 시스템 동작 환경도

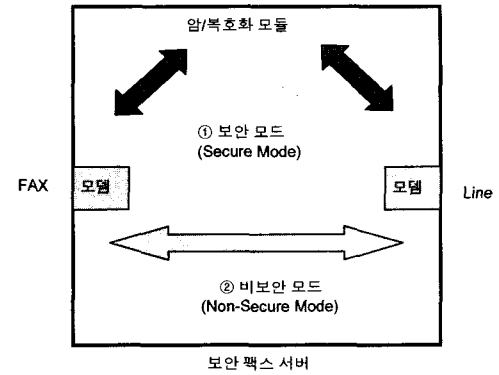
본 논문에서 구현된 시스템은 보안 모드와 비보안 모드 두 가지 모드를 지원하고 비보안 모드일 경우는 일반적인 팩스 간의 통신과 같다.

3.1 시스템 디자인

본 논문에서의 보안 팩스 서버는 기존 사용자가 팩스를 사용하던 환경의 변화를 최소화하면서 팩스 데이터를 안전하게 전달하고자 하였다. 즉, 사용자는 보안이 필요치 않은 문서의 경우는 일반적인 팩스 송수신 환경과 다를 바 없이 사용하고 보안용 문서라 할지라도 동작의 어려움 없이 팩스 송수신을 쉽게 할 수 있도록 하고자 했다. 또한 보안 모드로 팩스를 전송할 때라 할지라도 암복호화 시간의 오버헤드를 최소화해서 비보안 모드일 때의 시간에 근접하도록 하고자 했다.

이와 같은 요구사항을 만족하기 위해 본 시스템에서는 두 개의 팩스 모뎀을 사용하여 문서전송 시간을 최소화하였다. 즉, 팩스로부터의 문서를 송신하면서 다른 한 편으로는 암호화된 문서를 수신측의 팩스 서버로 전달하도록 하여 실시간 문서 전송을 하였다. 만일 하나의 팩스 모뎀을 사용하면 송신이나 수신 중에 다른 쪽으로의 수신이나 송신을 수행할 수 없기 때문에 전체 문서 전송시간은 2배 이상이 된다. 이는 사용자에게 심각한 불편 사항이 되기 때문에 본 시스템에서는 두 개의 모뎀을 사용하여 이를 해결하였다. (그림 2)는 이를 개념적으로 표현한 보안 팩스 서버의 내부 개념도이다.

두 개의 모뎀을 제어하기 위해서 프로그램 가능 논리 소자(Programmable Logic Device : PLD)를 사용하였고 초고속 집적 회로 하드웨어 기술 언어(VHSIC hardware description language : VHDL) 프로그램으로 제어하였다. 이에 대한 상세한 기술은 하드웨어 구현 절에서 다루며 개발된 시스템의 구성부품은 <표 1>과 같다.



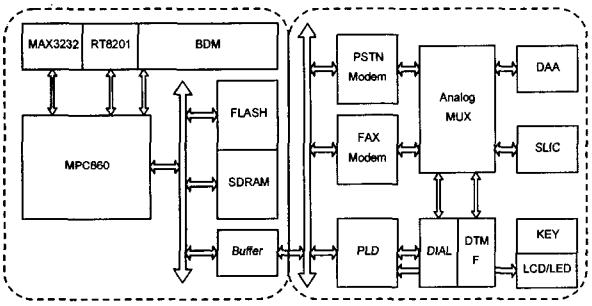
(그림 2) 보안 팩스 서버의 내부 개념도

〈표 1〉 시스템 구성 부품

구 분	회사 및 제품
마이크로프로세서	Motolora MPC855T 50MHz
메모리	8M 플래시, 32M SDRAM
모뎀	Conexant RC56D
DAA[1] (Data Access Arrangement)	CPCLARE CPC56
SLIC[2] (Subscriber-Line Interface Circuit)	AMD AM79R79-1J
FPGA	ALTERA EPM7128
운영체계	임베디드 리눅스 Kernel Ver 2.4.2

3.2 하드웨어 구현

전체 시스템의 하드웨어 블록도는 (그림 3)과 같다.



(그림 3) 하드웨어 블록도

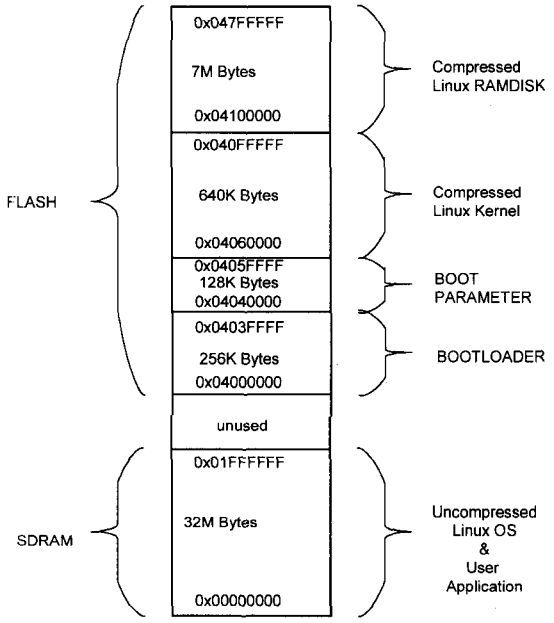
구현된 시스템의 하드웨어 부분은 몇 개의 기능별 블록으로 나누어진다. 이제 각 기능 블록 별 개발내용을 살펴본다.

1) DAA : 전화 라인 인터페이스 모듈
2) SLIC : 가입자선 인터페이스 회선

3.2.1 프로세서 블록

본 논문의 시스템은 50MHz로 동작하는 모토롤라 MPC-855T 네트워크 프로세서를 사용하였다. 내부에 명령어 캐시와 데이터 캐시를 각각 4Kbyte 씩 내장하고 있으며 직렬 통신과 이더넷 등을 포함하고 있다.

시스템을 구동을 위한 운영체계로는 임베디드 리눅스를 사용하였고, 운영체계를 위해 8Mbyte의 플래시와 32Mbyte의 동기식 동적 램(Synchronous Dynamic random Access Memory : SDRAM)을 사용하였다. (그림 4)는 시스템의 메모리맵을 나타낸 것이다.

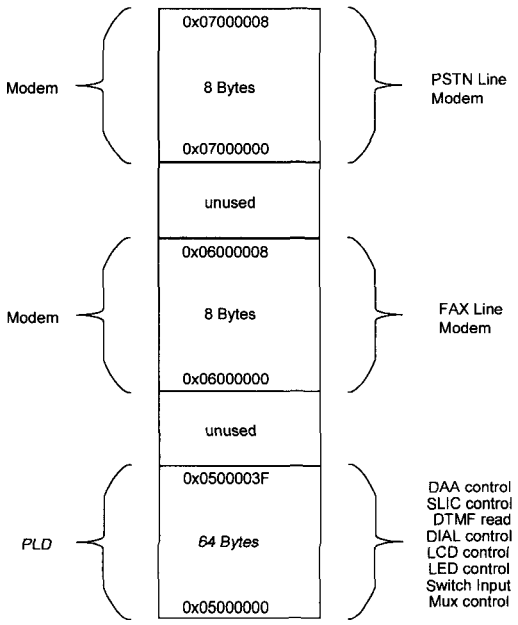


(그림 4) 시스템 하위 메모리맵

전체 8Mbyte의 플래시 영역을 4개의 영역으로 분할하여 사용하고 있다. 보드의 초기화 및 운영체계의 동작을 위한 TFTP(Trivial File Transfer Protocol : TFTP)[3] 다운로드 영역, 플래시 퓨징(Flash Fusing)에 요구되는 부트로더(Bootloader)와 리눅스 커널 부팅에 필요한 파라미터 저장 영역, 그리고 압축된 리눅스 커널 및 램 디스크 저장 영역으로 분할되어 있다.

부트로더(Bootloader)는 ppcboot-1.2를 기반으로 수정하였으며, TFTP 서버로부터 압축된 커널(약 340Kbyte)와 램 디스크(2.7Mbyte)를 동기식 동적 램에 저장하거나 플래시에 저장하는 기능을 가지고 있다. "bootm" 명령을 사용하여 동기식 동적 램에 저장된 커널 이미지로 부팅하거나 자동부팅 기능을 사용하여 플래시의 커널코드를 실행할 수 있다. 부트(boot) 파라미터 영역은 임베디드 리눅스 부팅을 위해 커널에 전달할 값들을 가지고 있으며, 이 영역에 IP 주소, 이더넷 MAC 주소와 부팅을 위한 선택사항들을 저장한다.

리눅스 커널과 램 디스크는 압축된 형태로 저장되어 있으며, 약 340Kbyte와 2.7Mbyte의 공간을 차지한다. 드라이버나 애플리케이션 개발 기간 동안 이더넷을 통하여 동기식 동적 램에 저장하여 사용하였으며, 개발 완료 후 플래시 메모리에 압축된 형태로 저장한다.



(그림 5) 시스템 상위 메모리맵

시스템은 팩스와의 입출력을 위한 팩스 모뎀과 일반 전화 교환망 선로 접속을 위한 모뎀을 내장하고 있다. 각 모뎀을 위한 메모리 공간을 시작번지 0x06000000과 0x07000000으로 할당하였으며 각각 8byte의 메모리 공간을 사용한다. 보드에 장치된 발광 다이오드(Light Emitting Diode : LED)와 액정 표시 장치(Liquid Crystal Display : LCD) 그리고 DAA/SLIC/아날로그 먹스/복합 주파수 부호(Dual Tone Multifrequency : DTMF)/ 다이얼과 관련된 신호의 입력과 제어를 위해 PLD를 0x05000000 영역에 할당하여 사용하였다.

3.2.2 모 뎀

● 팩스 모뎀

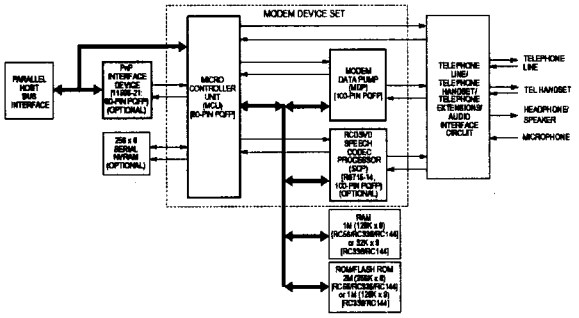
암호화하여 전송할 팩스 이미지의 수신이나 복호화된 팩스 이미지의 송신을 위하여 사용하며, 14,400bps까지 작동 가능하다. SLIC을 통하여 팩스와 직접 연결되며, 내부의 다이얼 톤 생성기와 복합 주파수 부호 감지회로를 사용하여 전화망에 연결된 것과 같은 동작을 수행한다.

● 일반 전화 교환망 모뎀

암호화된 팩스 이미지의 송수신을 위한 모뎀 통신에 사

3) TFTP는 FTP보다 간단하지만 기능이 조금 덜한 네트워크 애플리케이션이다. 사용자 인증이 불필요하고, 디렉토리를 보여주지 않아도 되는 곳에 사용되며 TCP 대신 UDP를 사용한다.

용되며, 외부 일반 전화 교환망과 DAA를 통하여 연결된다. 최대 57,600bps 수신이 가능하고, 38,400bps의 송신이 가능하다. 본 시스템에서 사용한 모드는 비동기 통신 모드에서 19,200bps로 고정하여 사용하였다. RC56D 모뎀 디바이스는 마이크로 제어 장치를 병렬 모드로 설정하여 사용하였으며, 256Kbyte의 롬과 128Kbyte의 램을 사용하였다. 모뎀 데이터 펌프(Modem Data Pump : MDP) 이외에 음성지원의 기능을 갖는 회로는 사용하지 않았으며, 마이크로프로세서와는 8비트 데이터 버스를 통하여 직접 연결된다.



(그림 6) RC56D 모뎀 블록다이어그램

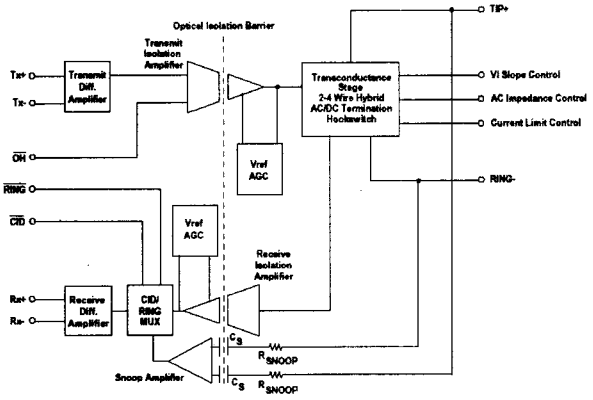
3.2.3 PLD

PLD에서는 다음과 같은 기능을 수행한다.

- DAA/SLIC/아날로그 먹스 제어신호 생성
- 복합 주파수 부호 detect 및 발신음 생성
- LCD/LED 제어 및 키 입력
- 일반 전화 교환망 모뎀/팩스 모뎀 제어 신호 생성

3.2.4 DAA/SLIC/아날로그 먹스(아날로그 신호)

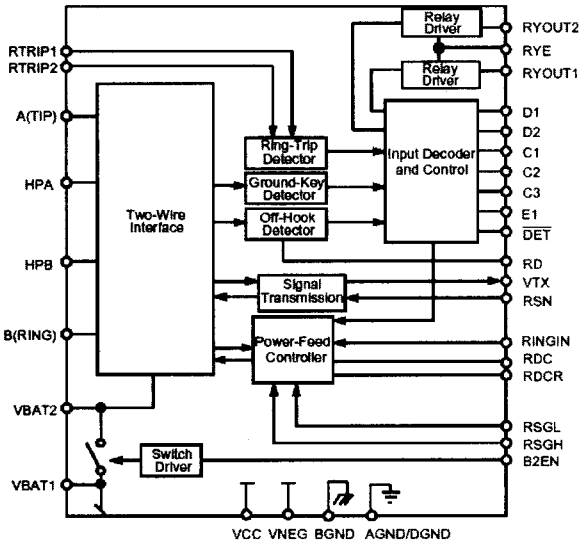
- DAA는 일반 전화 교환망의 링 팁(Tip) 신호로부터 아날로그 신호를 분리하여 아날로그 먹스로 전달하게 된다.



(그림 7) DAA 칩 블록다이어그램

입출력을 위한 RX/TX 신호와 응답 상태(off-hook)를 위한 OH#, 링 입력을 위한 RING#, 그리고 발신 번호 인식을 위한 CID# 포트를 가지고 있다. 아날로그와 일반 전화 교환망의 절연은 내부의 광 아이솔레이션 장벽(Optical Isolation Barrier) 를 통하여 구현되며, 1,500V까지 절연이 가능하다.

- SLIC은 호출 신호를 위한 회로를 내장하고 있다. 이를 위해 24V와 70V를 생성하는 회로가 필요하다. 호출 신호를 위한 20Hz를 PLD에서 생성하여 전달하며, 제어 신호 3개를 사용하여 SLIC의 동작 모드를 변경할 수 있다.



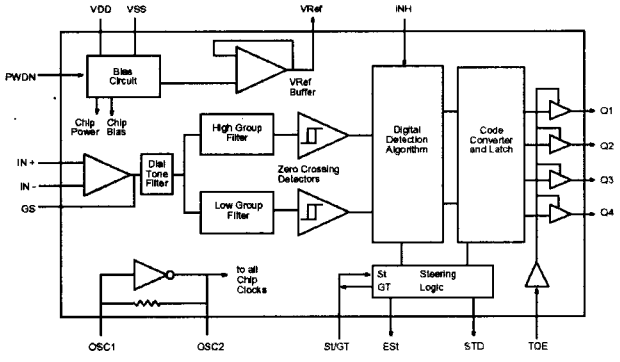
(그림 8) SLIC 칩 블록다이어그램

- 아날로그 먹스

아날로그 먹스는 DAA/SLIC/복합 주파수 부호/다이얼/모뎀들 간의 신호 전달을 위해 사용하여, 하나의 IC당 3개의 먹스를 갖는 4053을 사용하였다. 각각의 입출력은 약 2.4V를 기준으로 변화하며, 발신음 발생기의 경우 우회 콘덴서와 저항을 사용하여 전압 레벨을 변경하였다.

3.2.5 복합 주파수 부호 감지/발신음 생성

- 복합 주파수 부호 감지기

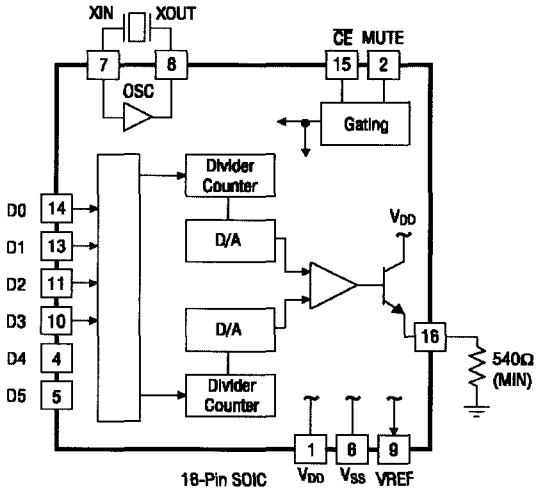


(그림 9) DTMF 블록다이어그램

CPCLARE 사의 MT88L70을 사용하였으며, 0~9, #, *의 입력을 감지할 수 있다. PLD와의 전달은 STD 핀의 logic '0' state에서 이루어지며, STD 핀의 하강 에지에서 복합 주파수 부호입력을 저장한다.

• 발신음 발생기

CPCPARE사의 M-991을 사용하여 발신음을 생성하며, 팩스 모뎀과 SLIC이 필요로 하는 응답 상태에서의 호출음과 다이얼 준비 톤을 생성한다.



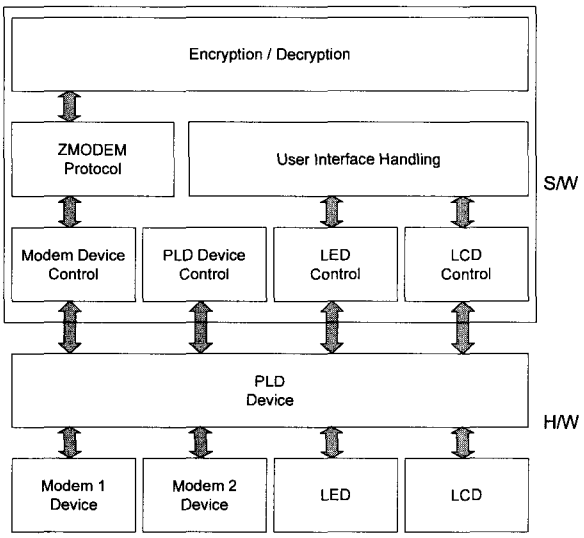
(그림 10) DTMF 톤 발생기 블록다이어그램

(그림 11)은 개발 완료 된 시스템 보드 사진자료이다.

(그림 11) 개발 시스템 보드 사진

3.3 소프트웨어 구현

본 절에서는 일반적인 임베디드 리눅스 개발과 관련된 커널 포팅 등의 내용은 언급하지 않고 운영체계 위에서 동작하는 본 시스템과 관련된 응용 애플리케이션 개발과 관련된 내용만을 기술한다. 본 시스템의 응용 애플리케이션 프로그램은 로컬 팩스와의 데이터 송수신 및 전화선을 통한 원격 팩스 서버와의 데이터 송수신을 담당하는 보안 팩스 서버 기능을 수행한다. 두개의 모뎀 디바이스와 PLD 디바이스를 제어하는 모듈이 각각 있고 보안모드일 경우 암복호화를 수행하는 모듈이 있다. 원격 팩스 서버와의 데이터 송수신 프로토콜은 Z-모뎀을 통해 구현되었다. (그림 12)는 기능별로 본 보안 팩스 서버의 구조이다.



(그림 12) H/W, S/W 블록 다이어그램

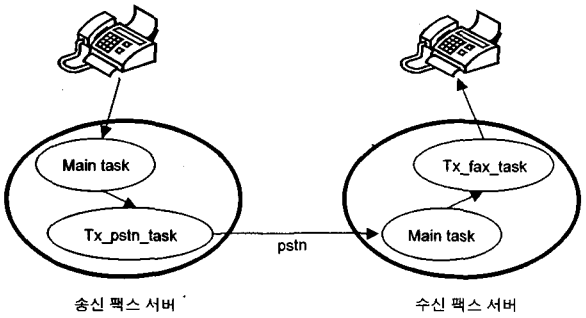
프로그램은 <표 2>와 같이 모두 3개의 태스크로 이루어져 동작한다. 처음 구동 시에 모뎀과 PLD 디바이스를 초기화하고 이벤트의 발생을 감시하는 메인 태스크와 송신측에서 원격으로 수신측에게 팩스 데이터를 내보내는 기능을 담당하는 태스크, 그리고 수신측에서 로컬로 팩스를 사용하여 데이터를 내보내는 기능을 담당하는 태스크로 이루어져 있다.

〈표 2〉 시스템 주요 태스크 이름과 기능

태스크 명	주요 기능
main_task	• 모뎀 및 PLD 디바이스 초기화 • 팩스나 전화선, 스위치들로부터 들어오는 이벤트 처리 • 로컬 팩스의 데이터 수신 • 리모트 팩스 서버의 데이터 수신 • 다른 태스크의 제어
tx_pstn_task	• 리모트 팩스 서버로 데이터 송신
tx_fax_task	• 로컬 팩스로 데이터 송신

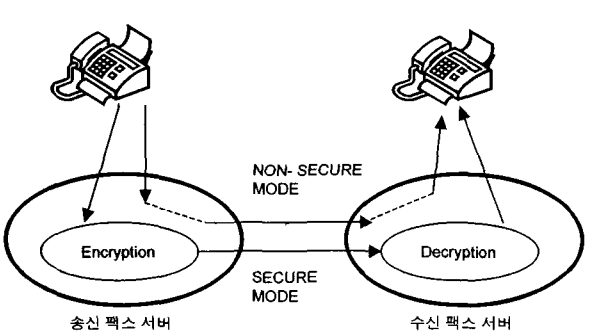
여러 개의 태스크를 만들어 처리한 이유는 2개의 모뎀을 동시에 구동하여 팩스 서버 내부의 데이터를 빠르게 처리하기 위함이다. 보안 팩스 서버는 지역의 팩스 데이터를 수신함과 동시에 리모트로 데이터를 송신할 수 있으며 반대로 리모트의 데이터를 수신함과 동시에 로컬로 데이터를 송신할 수 있다. 제어는 모두 메인 태스크가 담당한다. (그

림 13)은 태스크별로 본 팩스 서버의 구조를 나타낸다.



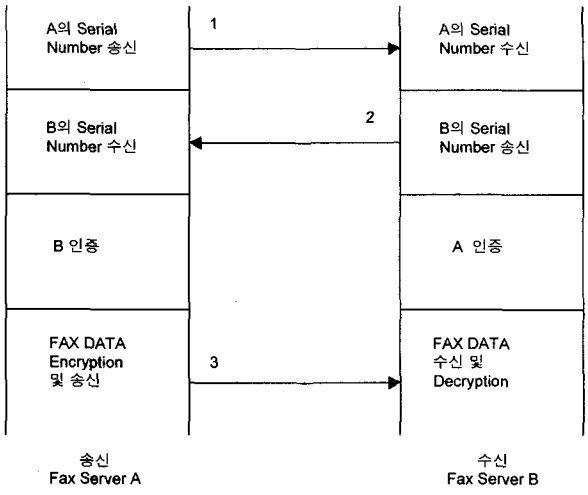
(그림 13) 송수신 팩스 서버의 태스크 동작 환경도

팩스 서버는 비보안 모드와 보안 모드의 두 가지 동작 모드를 갖고 있다. 초기화된 팩스 서버는 비보안 모드로 동작하며 보안이 필요할 경우 보안 모드로 동작시킬 수 있다. 동작의 전환은 팩스 서버의 스위치를 누름으로써 가능하다. 비보안 모드로 동작될 때 팩스 서버는 단지 로컬 팩스로부터 수신되는 데이터를 그대로 리모트로 전송하며 수신측 팩스 서버 역시 전송된 데이터를 그대로 로컬 팩스로 내보낸다. 이 과정에서 팩스 데이터에 대한 지연은 없다. 그러나 보안 모드로 동작될 때, 팩스 서버는 팩스 데이터에 대해 암호화를 처리하는데 암호화된 팩스의 데이터 송수신 과정은 다음과 같다. 수신측 팩스 서버는 걸려온 전화의 신호를 통해 일반 팩스에서 걸려온 전화인지 팩스 서버로부터 걸려온 전화인지 알 수 있는데 팩스 서버로 걸려온 전화일 경우 송수신간에 보안 모드로 동작하게 된다. 양측 팩스 서버는 먼저 시리얼 번호를 교환한 후에 서로 생성해낸 암호키를 교환한다. 송신측 팩스 서버는 자신의 암호키와 수신측 팩스 서버로부터 받은 암호키를 이용해 팩스 데이터를 암호화하고 암호화된 팩스 데이터를 수신측 팩스 서버로 전송한다. 수신측 팩스는 수신된 팩스 데이터를 이전에 교환한 두개의 암호키를 이용해 복호화 하여 본래의 팩스 데이터를 얻어낸다. 모든 데이터의 송수신을 마친 후에는 양측 모두 비보안 모드로 돌아간다. (그림 14)는 비보안 모드일 때와 보안 모드일 때의 팩스 서버의 동작을 나타낸다.



(그림 14) 보안 모드/비보안 모드 동작 개념도

(그림 15)는 위에서 설명한 보안 모드일 때의 팩스 서버의 동작 과정을 나타낸다.



(그림 15) 보안 모드의 동작 절차

데이터의 안전한 송수신을 위해 팩스 서버는 Z-모뎀 프로토콜을 사용하여 팩스 데이터를 교환한다. Z-모뎀은 많은 통신 프로그램에서 널리 사용되고 있으며 대량의 파일 전송이 가능하다. 수신측의 응답을 매번 기다리지 않고 송신측의 주도로 통신을 하여 데이터 처리 속도도 빠르며 CRC-32 에러 검출 방식을 채택해 통신의 정확성과 안정성이 높다. 또한 받은 파일에 대하여 이어받기가 가능하다는 장점이 있다

팩스 서버 프로그램은 보안 모드로 동작하여 통신할 때 서로의 시리얼 번호를 주고받는다. 시리얼 번호는 팩스 서버마다 고유하게 부여되는 13자리 문자열로 RZF-XXX-XXXXX 포맷을 따른다. 시리얼 번호는 램 디스크에 저장되지 않고 플래시 파일 시스템이라 불리는 플래시 메모리 상의 특정한 영역에 따로 저장되어 관리된다. 이를 관리하는 프로그램이 시리얼 번호 관리 프로그램의 목적이며 이 프로그램을 통하여 시리얼 번호의 변경이나 조회가 가능하다. 시리얼 번호 관리 프로그램은 다른 프로그램들과 마찬가지로 램 디스크 이미지에 저장되는데 이미지 디렉터리 내의 usr/bin/setrzf에 해당한다. 리눅스가 부팅된 후 콘솔 상에서 명령어 'setrzf'를 입력하여 시리얼 번호를 조회할 수 있고 파라미터로 시리얼 번호를 입력하여 시리얼 번호를 변경할 수 있다. 시리얼 번호를 변경한 후에는 시스템을 반드시 재부팅 할 필요는 없으나 변경된 시리얼 번호가 제대로 반영이 되었는지 확인하기 위해서는 시스템을 재부팅 해주는 것이 좋다.

4. 시험 평가

본 논문에서 개발한 보안 팩스 서버의 동작 성능을 시험

하기 위해 서버 없이 사용하는 일반적인 상황과 비교해 전송시간을 측정해 보았다. 비보안 모드의 경우는 일반 팩스와 동일한 전송 시간을 보였고 보안 모드일 때만이 전송지연이 발생하였다. 그러나 팩스와 팩스 서버 간에는 일반적인 팩스 송수신과 같은 전송속도로 데이터를 전송하지만 팩스 서버 간에는 모뎀통신을 통해 일반 팩스 송수신보다 빠른 데이터 전송을 수행하기 때문에 암복호화로 인해 발생하는 전송지연 시간을 상당부분 줄일 수 있었다. 본 시험은 물리적 위치를 이동해 가며 다른 전화국의 교환기를 사용하도록 시험해 보았는데, 이는 구축되어 있는 전화망의 상태가 지역별로 상이하여 모뎀 통신의 속도에 제한을 받을 수 있기 때문이다. 보안 모드에서 데이터 전송 시 접속이 원활하지 않으면 전송 속도를 낮춰 최대한 에러를 배제하려 하였다. 이 때문에 단 대 단 통신의 속도는 변할 수 있었으며 각기 다른 세 지역에서 시험을 수행했으며 한 지역에서 십여 차례 이상의 시험을 수행하고 그 평균을 계산하였다.

〈표 3〉 일반 팩스와 보안 팩스 서버의 전송시간 비교

전송 페이지 수	일반 팩스	보안 팩스 서버
1장	35초	100초
2장	72초	143초
3장	110초	175초
4장	150초	226초

연결 지연시간은 배제하고 순수하게 데이터 전송 시간만을 계산하였다. <표 3>와 같이, 보안 팩스 서버를 거치지 않고 전송하는 경우 전송 페이지 수에 관계없이 일정 시간만이 증가함을 알 수 있다. 즉 보안 팩스 서버를 거쳐 암복호화를 수행하더라도 단지 2페이지 정도의 전송지연만이 발생함을 알 수 있다. 이는 팩스로부터 첫 페이지를 수신한 시스템에서 다음 페이지를 수신하는 동안 암호화를 수행한 후 또 다른 모뎀을 통해 팩스에 비해 빠른 모뎀 데이터 통신을 수행하여 송신하기 때문에 두 번째 페이지를 받기 이전에 암호화된 페이지의 송신은 모두 완결되기 때문이다. 이는 팩스로부터 팩스 서버로, 팩스 서버간의 전송, 마지막으로 팩스 서버로부터 팩스로의 3단계의 전송을 거치기 때문에 생기는 지연이다. 페이지 수가 증가하더라도 항상 상수 시간만큼의 지연이 발생한다는 것을 알 수 있으며, 전체 전송 시간으로 볼 때 이 정도의 지연은 사용자가 충분히 견딜 수 있는 수준이다.

5. 결론 및 향후연구

본 논문에서는 일반 기업에서 데이터 전송의 중요한 수단으로 사용되는 팩스의 보안 기능을 높이기 위해 임베디드 보안 팩스 서버 개발에 대해 기술하였다. 인터넷이라는 새로운 통신 수단의 출현 이후에도 여전히 팩스는 중요한 데이터 전송 수단이지만, 그 보안성에 대한 인식은 부족한 듯하다. 본 논문의 보안 팩스 서버는 기존의 사용 환경의 변화 없이 쉽게 적용 가능하며 손쉽게 사용 가능하도록 하였다.

비보안 모드와 보안 모드의 두 가지 모드를 지원하여, 비보안 문서의 경우 평상시 사용할 때와 똑같이 문서전송이 가능하다. 보안 모드는 보안 문서 전송 시 사용되며 비보안 모드 경우와 비교하여 전체 전송 시간은 평균 두 페이지 정도의 지연만이 있을 뿐이다.

본 논문에서는 보안 팩스 서버 개발의 하드웨어, 소프트웨어 개발에 대한 전반적인 내용을 다루었으며, 시스템을 동작 시험한 결과, 기능 및 성능 면에서 산업화 가능성을 보였다.

시스템의 암호 모듈은 국내 표준을 구현하여 국내에서 사용하기에 충분한 조건을 갖추었으며 암호 동작 성능 역시 검증되었다. 현재의 보안용 팩스는 국제 표준을 따른 것이 없으며 각자 독자 기술로 구현되었다. 현재 시스템에서 키 분배 시 우려되는 보안 유지를 위해 보안성을 높일 수 있는 방안과 여러 공격에 대한 시뮬레이션을 통해 안전성에 대한 검증을 확보해야 할 것이다.

참 고 문 헌

[1] K. McConnell, D. Bodson, and S. Urban, 'Facsimile Technology and Systems', 3rd Ed., Artech House, 1999.

[2] 박영환, 임베디드 시스템 & 임베디드 리눅스, pp.2-26, 2002.

[3] C. O. Angaye, "Security in a networked environment", ACM SIGAPP Applied Computing Review archive, Vol. 3, No.1, pp.2-5, 1995.

[4] 이종수, "G4 FAX 개발현황", 정보통신 한국통신학회지, Vol.10, No.4, pp.36-42, Apr., 2002.

[5] 원제혁, "퍼스털 컴퓨터팩시밀리 (PC-FAX)의 소개", 정보통신 한국통신학회지, Vol.5, No.2, pp.197-202, June, 1988.

[6] 강민구, 김종일, 강창언, "공중전화망(PSTN)을 이용한 PC-FAX통신에 관한 연구", 한국통신학회 추계학술발표회 논문집, pp.44-47, 1988.

[7] A. Margolis, 'The Fax Modem Sourcebook', John Wiley & Sons, 1995.

[8] 이만영 외 5인, '현대 암호학 및 응용', 생능출판사, 2002.

[9] IETF RFC2542, Terminology and Goals for Internet Fax, Mar. 1999.

[10] http://www.enfax.co.kr.

[11] http://www.magicfax.co.kr.

[12] http://www.hylafx.org.

이 상 학

e-mail : shlee@keti.re.kr
1993년 전주대학교 수학과(이학사)
1997년 경희대학교 대학원 컴퓨터공학과 (공학석사)
2000년 경희대학교 대학원 컴퓨터공학과 (박사수료)
2000년~현재 전자부품연구원 유비쿼터스컴퓨팅연구센터 선임연구원
관심분야 : Sensor Network, Combinatorial Optimization, Meta-Heuristic Algorithm

정 태 충

e-mail : tcchung@khu.ac.kr
1980년 서울대학교 전자공학과(공학사)
1982년 한국과학기술원 대학원 전자계산공학과(공학석사)
1987년 한국과학기술원 대학원 전자계산공학과(공학박사)
1987년~1988년 KIST 시스템 공학센터 선임연구원
2001년 미국 Iowa 대학 교환교수
1988년~현재 경희대학교 컴퓨터공학과 정교수
관심분야 : 인공지능, 지능에이전트, 메타알고리즘