Visual C++을 이용한 윈도우 운영체제 내의 파일 및 디렉토리 보안 기능 설계

Design of files and directories with security features within the Windows O.S using Visual C++

장승주, 김준호
동의대학교 컴퓨터공학과

Seung-Ju Jang, Kim Jun-ho
Dept. of Computer Engineering, Dong-Eui University

요약

이 프로그램은 Visual C++로 개발되었으며, 윈도우 운영체제 내에서 파일 및 디렉토리 보안 기능을 가지고 있다. 파일 및 디렉토리 보안은 암호화/복호화 작업으로 이루어지며 파일 보안은 키 값과 라운드수를 알아야 하고, 디렉토리 보안은 비밀번호를 알아야만 복호화 할 수 있도록 설계되었다. 또한 ECB, CBC 연산방식과 3DES, SEED 알고리즘방식을 지원하며, 암호화 시 .de0 이라는 실행할 수 없는 확장자의 파일로 만들어지면서 이중보안을 할 수 있도록 개발되었다.

Abstract

This program was developed in Visual C + +, the Windows operating system has security features within the files and directories. File and directory security, encryption / decryption operations yirueojimyeo file security can be round, to know the value of the key and security password I need to know the directory is designed to be decrypted. In addition, ECB, CBC algorithm and 3DES, SEED algorithms and methods, and encryption. De0 can not run that created the file extension, as has been developed to allow for double security.

Ⅰ. 서론

정보통신기술은 날이 갈수록 계속 발전하고 있다. 정보통신기술이 나날이 발전하면서 그만큼의 편리함도 있지만 정보유출 또한 많이 일어나면서 자신의 컴퓨터에 있는 자료도 안전 할 수 없는 상황이 되었다. 그러나 현재 거의 대부분의 PC사용자들을 보면 컴퓨터 바이러스에 대한 대비는 많이 하면서도 타인이 자신의 정보를 허가 없이 유출해가는 것에 대한 대비는 매우 부족한 편이다. 지금 출시되고 있는 제품만 보더라도 바이러스를 치료하는 프로그램은 많아도 정보의 유출을 막을 수 있는 프로그램은 거의 찾아볼 수 없을 정도이다. 이렇듯 프로그램도 많이 없고 사람들의 인식 또한 부족한 상황이다. 이런 잘못된 현실을 바로잡고자 하여 파일보안 및 디렉토리 보안 프로그램을 개발하였다.

본 논문은 총 5장으로 구성되어 있으며 2장에서는 파일 및 디렉토리 보안 관련 연구에 대해서 기술하였고, 3장에서는 파일 및 디렉토리 보안 기능 설계에 대해서 기술하였다. 4장에서는 기능을 테스트하였고 마지막으로 5장에서는 결론 및 앞으로의 연구방향에 대해서 기술하였다.

Ⅱ. 관련연구

프로그램을 개발하는데 앞서 파일보안과 관련된 알고리즘에 대하여 조사하였다.

1. SEED

SEED 암호화 알고리즘은 8, 16, 32비트의 데이터처리가 모두 가능하고 블록암호방식을 사용한다. 입출력문의 크기는 128비트, 입력키의 크기도 128비트이다. DC/LC에 대하여 안전하도록 설계되었으며, Feistel 구조를 가진다. 내부함수는 SPN 구조이며, 비선형함수를 Look-up 테이블로 변형하여 사용한다.

라운드 수를 정함에 있어 안전성은 키 전수 조사공격에 필요한 계산복잡도 및 평문 암호문 쌍(2128) 이하가 되지 않아야 하며, 효율성 요구조건을 만족하여야 한다. 키 생성 알고리즘의 라운드 동작과 동시에 암/복호화 라운드 키가 생성될 수 있도록 설계되었다.

안전성이 증명 가능한 구조로 설계되어 있다. 차분해독법(Differential Cryptanalysis, DC), 선형해독법(Linear Cryptanalysis, LC)에 대하여 안전하여야 한다.

기타 공격방식(Higher Order DC 등)이 적용되기 어렵게 하기 위하여 대수적 차수가 3 이상인 부울함수를 사용한다. Related Key Attack에 강하기 위하여 Key Schedule에 비선형 함수를 사용한다. 효율성에 대한 설계조건은 S/W로 구현 시 3중 DES보다 고속이어야 한다.

기본적으로 SEED는 민간분야의 암호사용을 촉진하기 위하여 개발된 암호 알고리즘이다. 따라서, 개인 및 기업에서의 중요정보를 보호하기 위하여 필요한 경우 SEED 사용과 관련해서는 아무런 제약이 없다.

가장 대표적인 활용분야로는 현재 활발히 추진되고 있는 인터넷을 이용한 전자상거래 분야이다. 특히, 전자서명법과 관련하여 1999년 7월 1일부터 공인인증기관의 운영 및 전자서명의 법적 효력이 부여될 예정으로 전자상거래가 매우 활발히 촉진될 것으로 기대된다. 그러나 전자서명법에는 직접적으로 암호사용과 관련한 부분이 없으며, 전자상거래의 안정성, 신뢰성 확보를 위해서는 공개키 인증과 더불어 암호사용이 필수적으로 수반되어야 할 것이다. 이 경우 중요 정보의 보호를 위해 표준 암호 알고리즘인 SEED를 사용하도록 권고하고 있는 것이다. 구체적인 사용 예로는 전자상거래 이용 시 전자거래 내용 및 계좌번호 등의 노출방지를 위한 암호화 등이 있을 것이다. 한편 전자화폐나 전자지불시스템 구현 시에도 아무런 제약 없이 필요한 데이터 암호화에도 적용이 가능하다. 또한 전자 상거래 외에 대표적인 활용분야는 다음과 같다. 전자 우편 시스템에서의 메시지 암호화, PC의 저장된 데이터의 암호화, 가상교육 시스템, 유료 수업 내용의 보호, 개인별 성적표 내용 등의 보호, 위성방송사업과 연관된 한정수신시스템(CAS : Conditional Access System), 유료 방송 내용의 암호화 등이다.

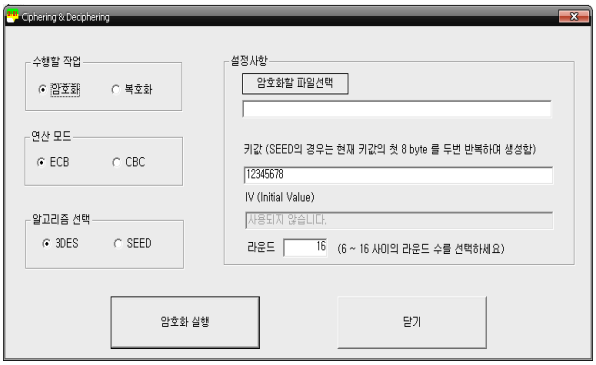
2. 3DES

3DES 암호화 알고리즘은 세계적인 표준으로 사용된 64비트 블록 암호 알고리즘이다. DES의 구조는 데이터 암호부와 키 생성부로 구성되어 있다. 먼저 키 생성부에서 생성된 48비트의 16개 라운드 키는 데이터 암호부의 각 라운드로 들어가 평문 블록과 함께 치환, 대치, 키 스케줄 등을 통하여 암호문을 만들어 내고, 복호화는 암호화의 역순이다.

기존의 DES 알고리즘이이 키 길이가 짧아 해독이 쉬운 단점을 극복하기 위하여 개발된 것으로 DES 알고리즘을 암호-복호-암호의 과정으로 연달아 적용해서 보안성을 강화한 것이다. 3DES 알고리즘은 기존의 DES 암호화 알고리즘보다 키 길이가 3배나 길어 깨기 힘들다.

Ⅲ. 파일 및 디렉토리 보안 기능 설계

파일 및 디렉토리 보안 프로그램은 회원가입을 통한 로그인을 해야만 실행할 수 있으며 암호화/복호화 방식으로 이루어진다. 연산방식은 ECB와 CBC를 사용하고, 알고리즘은 3DES와 SEED를 사용한다. 폴더 암호화는 암호화/복호화에 비밀번호를 설정하므로 비밀번호를 알지 못하면 복호화를 할 수 없게 되고, 파일 암호화는 키 값과 라운드수를 사용자 임의로 선택할 수 있으며, 이 역시 키 값과 라운드수를 알지 못하면 복호화를 할 수 없게 된다. 또한 암호화를 하게 되면 de0이라는 실행할 수 없는 파일로 변환되므로 한층 강화된 보안을 할 수 있다. 다음 그림 1은 파일 보안을 실행했을 때 모습이고, 그림 2는 디렉토리 보안을 실행했을 때 모습이다.



▶▶ 그림 1. 파일 보안 실행화면

▶▶ 그림 2. 디렉토리 보안 실행화면

파일 보안은 '수행할 작업' 에서 암호화 할 것인지 복호화 할 것인지를 선택하고 '연산모드' 에서 연산방식을, '알고리즘 선택' 에서 사용할 알고리즘을 선택할 수 있다. '암호화 파일 선택' 에서는 암호화 하고자 하는 파일을 찾을 수 있으며 키 값과 라운드 수를 설정함으로써 복호화 시 비밀번호 기능을 수행하게 된다. 앞서의 '수행할 작업' , '연산모드' , '알고리즘 선택' 에 따라서 키 값, 라운드 수의 옵션 또한 변경된다.

디렉토리 보안은 그림 2에서 좌측의 폴더를 선택하여 '암/복호화에 사용할 비밀번호' 란에 비밀번호를 입력한 후 '폴더 암호화' 버튼을 누르면 암호화가 시작된다. 복호화 시에는 암호화된 폴더를 선택하고 암호화 시에 사용된 비밀번호를 입력한 후 '폴더 복호화' 버튼을 누르면 복호화가 된다.

알고리즘(3DES, SEED) 소스는 다음과 같다.

표 1. 알고리즘(3DES, SEED) 소스

```
if(암호화 체크여부 확인){
    if(3DES 알고리즘 체크여부 확인){
        3DES 알고리즘 실행
    }
    else{
        SEED 알고리즘 실행
        }
```

3.1 프로그램의 구성

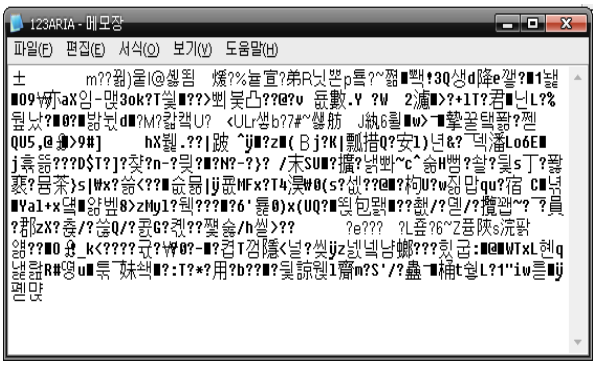
이 프로그램은 크게 파일 암호화/복호화, 디렉토리 암호화/복호화 작업으로 나뉘어 지며 로그인 시스템으로 프로그램이 실행되어 진다. 로그인 시스템은 Microsoft Office Access를 이용하여 연동 되어 지므로 이를 연동하기 위해서는 제어판의 관리도구 에서 이 프로그램과 Microsoft Office Access를 연결시켜 주어야 한다. 그리고 이 프로그램은 Intel Pentium4 3.0GHz CPU, 1GB Memory, Windows XP OS 에서 구현되었으며 이 프로그램을 구성하기 위하여 Microsoft Visual Studio 6.0 과 Microsoft Office Access 2003 이 사용되었고, 프로그램을 실행하기 위해서는 약30MB 의 하드용량을 필요로 한다.

Ⅵ. 실험

3장에서 설계한 파일 및 디렉토리 암호화/복호화 시스템을 테스트하였다.

1. 디렉토리 암호화 및 복호화

그림 3은 암호화할 폴더를 선택하고 비밀번호를 입력한 후 정상적으로 '폴더 암호화' 버튼을 눌렀을 때의 실행화면이다. 이와 같이 암호화가 정상적으로 이루어지면 파일 내용을 확인 할 수 없게 되는 것을 볼 수 있다.



▶▶ 그림 3. 암호화 파일 실행화면

디렉토리 복호화는 암호화 시 입력한 비밀번호를 입력한 후 '폴더 복호화' 버튼을 누르게 되면 동일한 값인지를 확인한 후 일치한 경우만 복호화를 실행한다. 다음 그림 4는 복호화를 했을 때 암호화 시 입력했던 비밀번호와 일치했을 경우의 모습이다.

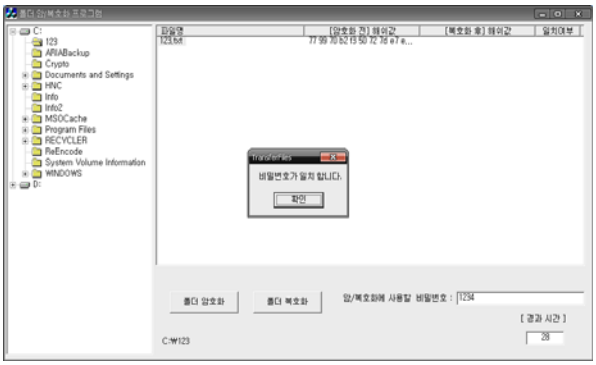
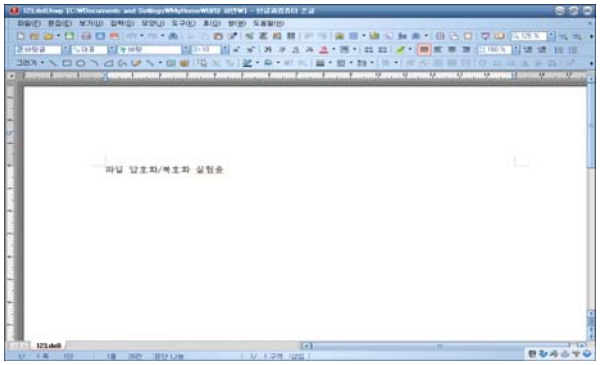


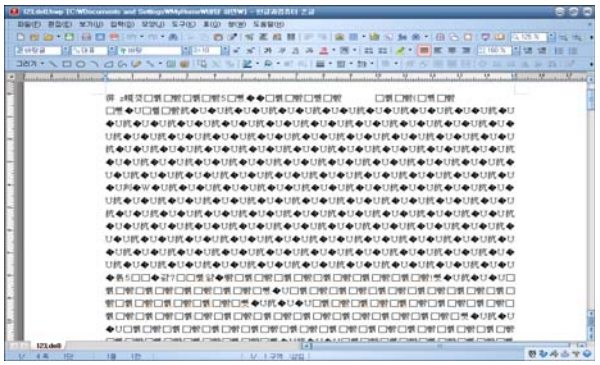
그림 4. 복호화 실행화면

2. 파일 암호화 및 복호화

먼저 암호화를 수행하기 위해 키 값 '13579246', 라운드 수 '7' 을 주고 복호화 시 암호화 시와 같은 값을 주었을 때와 다른 값을 주었을 때의 실행결과를 비교하였다.



▶▶ 그림 5. 같은 값의 복호화 실행결과



▶▶ 그림 6. 다른 값의 복호화 실행결과

그림 5는 암호화 시와 같은 값을 주었을 때의 복호화 실행결과 이고, 그림 6은 암호화 시화 다른 값을 주었을 때의 복호화 실행결과 이다. 그림 6에서 볼 수 있듯이 복호화 시 암호화 시와 다른 키 값과 라운드수를 주게 되면 복호화 결과가 올바르게 나오지 않음을 알 수 있다.

Ⅴ. 결론

오늘날 PC 사용자들은 바이러스 대비에만 신경을 쓰고 보안에는 신경을 쓰지 않아 보안프로그램을 개발하였다. 이로 인해 사용자들은 PC내의 중요문서를 안전하게 보관할 수 있게 되었고, 정보 유출을 방지할 수 있게 되었다. 앞으로의 연구방향은 이 프로그램의 암호화/복호화로 인해 발생할 수 있는 문제점들에 대해서 살펴보고 기존 연구들과 비교, 분석 하면서 이 문제점들을 해결하고 보안에 필요한 여러 가지 기능들을 더 추가

할 예정이다.

■ 참 고 문 헌 ■

[1] 정일홍, “MFC로 구현한 윈도우 프로그래밍”, 생능출판사, 2006
[2] 김용성, “Visual C++ 6.0 완벽가이드 “, 영진출판사, 2004
[3] 강선명, “Visual C++ 암호화 프로그래밍”, FREELEC
[4] 최호성, “Visual C++ 2008 MFC 윈도우 프로그래밍”, FREELEC, 2008
[5] 한국정보보호 진흥원, www.kisa.or.kr/index.jsp
[6] 신화선, “윈도우 프로그래밍 Visual C++ MFC Programming”, 한빛미디어, 2003
[7] 윤성일, “C++ 기초플러스 5판”, 성안당, 2006
[8] 알고리즘(SEED, 3DES)관련, www.securitytechnet.com/std-algorithm/block.html#block1
[9] 양재찬, “정품 소프트웨어가 들어있는 C++", 정보문화사, 2004
[10] 성윤정, “C++ 프로그래밍 기초 객체지향의 시작”, 한빛미디어, 2007
[11] 이경휘, “단계별 실습으로 배우는 Visual C++ 6.0", 생능출판사, 2002