通信情報保護學會論文誌
第9卷 第1號, 1999. 3

MPCOS-3DES를 이용한 인증 시스템의 구현

김 경훈*, 박 창섭**

Implementation of Authentication System with MPCOS-3DES Smartcard

KyeongHun Kim*, ChangSeop Park**

요 약

본 논문에서는 스마트 카드를 이용한 인터넷 환경에서의 클라이언트-서버 인증 시스템을 설계, 구현하였다. 대칭형 암호인 삼중 DES를 지원하는 GEMPLUS사의 MPCOS-3DES 스마트 카드를 통해서 시도-응답(Challenge-Response) 기반의 사용자-카드, 클라이언트-서버 그리고 카드와 서버간의 3단계 인증 과정이 수행되어 진다.

Abstract

In this paper, the Client-Server authentication system is proposed and implemented using the smart card on the internet. Based on the MPCOS-3DES smart card manufactured by GEMPLUS, three phases of authentication using the challenge-response mechanism are performed, which includes user-card authentication, client-server authentication, and card-server authentication.

Key word : Smart card, MPCOS, Challenge-Response, DES, Authentication

1. 서 론

정보 통신 시스템의 발달 및 인터넷의 사용 증가로 인하여 통신 시스템 및 컴퓨터망을 통한 전자 상거래의 발전은 새로운 인증 시스템의 개발과 안전성이 높고 사용하기 편리한 새로운 메커니즘을 요구한다.[5] 현재, 통신망을 통한 신용카드 기반의 상거래는 마그네틱 카드 번호를 이용하여 거래가 되며, 카드의 복제

* 고려대학교 전산학과
** 단국대학교 전자계산학과

와 통신상에서 제 3자에 의해 도청되어 질 수 있다는 단점이 있다. 또한, 각 은행과 통신망을 이용한 상거래에서 개인 신분 확인을 위한 패스워드 방식은 사용자로 하여금 각 은행과 통신망 접속에 필요한 패스워드 관리와 유출 방지에 대한 부담을 주고 있는 것이 현실이다. 이를 보완하기 위해 ICC(Integrated Circuit Card)라는 신용카드 크기에 마이크로프로세서와 메모리, 입출력 장치 등을 내장한 카드가 개발되었다. ICC는 자신만의 운영체제 즉, COS(Card Operating System)를 갖고 있으며, COS는 외부와 통신 시에 암호 시스템을 사용하여 제 3자의 공격으로부터 안전하게 보호하여 준다. 또한, 카드에 저장된 파일 시스템의 보호를 위해 파일 헤더 부분에 제어조건을 첨부하여 자료에 대한 무결성을 제공한다.[2][3][6] 본 논문에서는 대칭형 암호체계를 지원하는 스마트 카드를 사용하였고, 최근에는 공개키 암호를 지원하는 ICC도 발표되어지고 있다.[4]

본 논문에서는 GEMPLUS 사의 MPCOS-3DES(Multi Purpose Card Operating System-3DES) 카드를 이용하여 원격지에 있는 서버에 접속하기 위해 3 단계로 인증 과정을 구현하였다. 2 장에서는 MPCOS-3DES의 파일 시스템과 저장된 자료의 무결성을 지원하는 방법에 대해 기술하고, 3 장에서는 본 논문에서 제안한 인증 프로토콜의 설계를 기술한다. COS의 파일 시스템의 구조 설계와 개방형 네트워크에서 양자간 통신시에 자료의 노출을 방지하기 위해 대칭형 암호인 DES(Data Encryption Standard)와 3중 DES를 사용하고, 3 단계의 인증 과정을 제안한다. 1 단계는 사용자와 카드간의 인증으로 사용자가 사용자 인증 값과 가상키를 입력하여 정당한 사용자임을 인증 받게 되고, 2 단계는 서버와 클라이언트간 세션키 공유과정을 통해 인증 과정을 수행하며, 3 단계는 서버와 카드간 인증으로 카드에 저장된 서버 인증 값 파일을 클라이언트가 읽을 수 있도록 필요한 정보를 서버쪽에서 전송함으로서 이루어진다. 4 장에서는 3 장에서 기술된 프로토콜의 구현 결과에 대해 기술하고, 5 장에서는 결론과 향후 연구 방향을 제시한다.

2. MPCOS-3DES 스마트 카드

MPCOS-3DES 카드는 ISO-7816에 규정한 스마트 카드의 일반적인 구성과 표준을 따른다. 스마트 카드는 8 비트 마이크로프로세서와 COS가 저장될 ROM, 임시 기억장소인 RAM, 64 KB인 EEPROM, RESET 회로, Input/Output, Clock회로를 갖고 있다. EEPROM에 저장된 자료는 오직 COS에 의해 제어가 가능하며 터미널과 카드간의 전송은 DES(Data Encryption Standard)나 3중 DES를 이용한 암호화가 이루어진다. MPCOS-3DES 카드의 운영체제는 ISO-7816/4에서 규정한 자료의 구조, 명령어 형식, 명령어 수행에 대한 리턴 값으로 이루어진다.[6][7][8]

2-1 MPCOS-3DES 파일구조

파일의 구조는 글로벌 단계와 로컬 단계로 이루어진 계층적 구조를 갖고, 최상위에 MF(Master File) 그리고 하위에 DF(Dedicated File)와 EF(Elementary File)를 갖는다. 그림 2.1은 COS 파일 시스템의 예이다. 일반 컴퓨터의 계층적 파일 시스템의 경우와 비교한다면 MF는 루트와 같고 DF는 디렉토리, EF는 파일과 같다. 모든 파일은 16 바이트의 파일 표시자와 파일 내용으로 구성되는데 MF의 파일내용은 없고, DF의 경우 DF의 이름으로 구성된다. EF는 파일의 형식에 따라 서로 다른 크기와 내용으로 구성된다.[7][8]

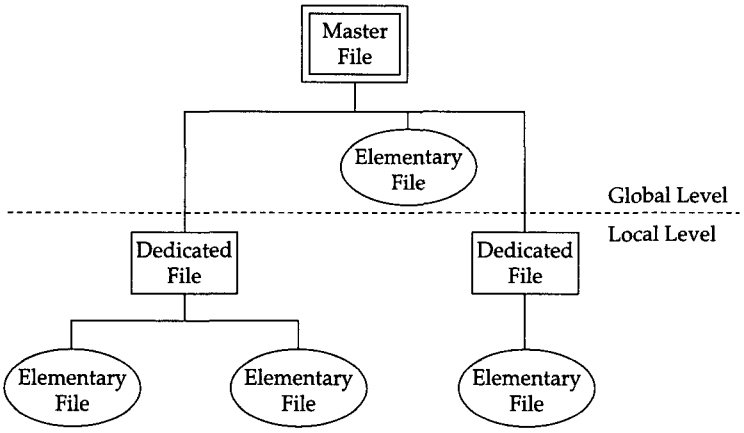


그림 2.1 MPCOS-3DES의 파일시스템
Fig 2.1 File structure of MPCOS-3DES

FP	Fn	Identification	
FDB	OPT	CYPTO	BodySize
Group 1 AC		Group 2 AC	
00	00	Status	CheckSum

FP	Fn	Identification	
FDB	OPT	BodySize	
Group 1 AC		Group 2 AC	
Group 3 AC		00	CheckSum

그림 2.2 파일 표시자
Fig 2.2 File descriptor

그림 2.2는 파일 표시자의 구조이며, 운영체제에 의해 할당되는 FP(File Pedigree), Fn(File number), 사용자나 프로그램에서 정의하는 Identifier, 파일의 유형을 표시하는 FDB((File Descriptor Byte), 파일의 크기를 위한 Body Size, 파일 제어를 위한 조건을 정의 할 수 있는 Group nAC(Access Condition)필드 그리고 Checksum 바이트를 제외한 나머지 바이트를 배타적 논리합(XOR)을 수행한 결과가 저장되는 Checksum 바이트로 구성된다. DF의 CRYPT 두 비트는 암호화 방식 결정에 관한 정의 필드로 DES 혹은 3중 DES로 설정 할 수 있다. 파일의 내용은 DF의 경우에는 DF의 이름으로 구성되고, EF에는 파일의 유형에 키 파일, 비밀 코드 파일, 사용자가 정의하는 일반 파일에 따라 일정한 크기와 형식을 갖는다. 키 파일은 12 바이트로 4 바이트의 헤더와 8 바이트의 DES 키가 저장되며, 3중 DES의 경우 두 개의 DES 키 값을 사용한다. 키의 종류에는 관리(Administration) 키, 지불(Payment) 키, 서명(Signature) 키가 있으며 관리키는 파일의 읽기, 쓰기, 수정과 같은 ISO 명령어를 위한 세션 키를 만드는데 사용되고 지불키는 지불 시스템에서 지불 세션 키를 생성에 사용된다.

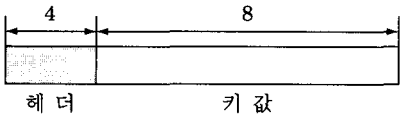


그림 2.3 키파일의 구조
Fig 2.3 Structure of key file

그림 2.3은 키 파일의 구조이다. 헤더에는 키의 유형을 정의하는 Type 필드와 키 파일의 버전에 관한 Kv, Cks(Checksum)는 Cks를 제외

한 나머지 바이트를 배타적 논리합(XOR) 연산 수행에 대한 결과로 구성된다. 키 값에는 8 바이트 DES 키가 저장된다. 하나의 키 파일에는 여러 개의 키들이 저장되며, 파일 내에서 키의 식별은 상대 주소를 통해 갖는 12 바이트를 하나의 단위로 저장하게 된다. 3중 DES의 상대적인 키의 번호는 0,2,4... 순으로 부여한다.

MF나 DF는 하나의 비밀코드 파일에 8 개의 비밀코드를 저장할 수 있으며, 4 바이트의 헤더와 4 바이트의 비밀코드로 구성된다. 헤더 부분에는 비밀코드의 입력 오류 횟수가 저장된 SCR(Secret Code Ratification), 터미널에서 오류 비밀코드를 보낼 경우 입력 횟수를 제한하는 MPN(Maximum Presentation Number), 잠긴 파일을 해제 할 수 있는 UCR(Unlock Code Reference)로 구성된다. 그림 2.4는 비밀코드 파일의 구조와 비밀 코드 값을 도출하는 과정을 도식화 한 것이다.

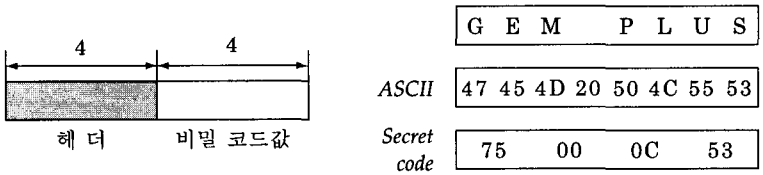


그림 2.4 비밀코드 구조와 추출의 예
Fig 2.4 Structure and Example of secret code value

2-2. MPCOS-3DES의 보안 메커니즘

MPCOS-3DES 카드의 보안 메커니즘은 카드 내에 저장된 자료의 보호와 카드와 터미널간 암호화를 통한 전송에 있다. 자료 보호를 위해서 파일 표시자에 명시된 제어조건 그룹을 통해 이루어지고, 카드와 터미널간 암호화 전송은 대칭형 암호인 DES 혹은 3중 DES를 통해 세션키를 공유하고 이 키를 이용하여 암호문을 생성해서 전송한다.

제어조건은 터미널에서 파일의 내용을 요구할 경우 COS에 의해 처리된다. DF의 경우 두 개의 제어조건 그룹을 갖는다. 첫 번째는 Common File -COS에서 사용되지 않는 일반 파일- 에 관한 제어조건이고, 두 번째는 Sensitive File -규정된 파일 : MF, DF, 키 파일, 비밀코드 파일- 에 대한 제어조건이다. EF의 제어조건 그룹은 파일의 수정, 쓰기 그리고 읽기에 대해 각각 2 바이트로 구성되며, 해당 제어조건 그룹에는 키 파일과 비밀코드 파일의 번호를 갖고 있으며 키는 카드와 터미널간에 전송되는 자료의 암호화에 사용되고, 비밀코드는 파일 제어 권한을 요구한 터미널로부터 COS가 입력받은 터미널의 비밀코드 값과 비교하여 권한을 부여한다.

카드와 터미널간 전송에 사용되는 암호체계는 대칭형 암호 방식인 DES와 3중 DES를 이용한다. 키의 비밀성을 유지하기 위해 세션 키를 사용하는데 세션 키 공유 과정에서 카드와 터미널간 인증 과정이 수행되는 것이다. MPCOS-3DES의 인증은 내부 인증과 외부 인증으로 분류되는데 내부 인증은 카드가 자신이 정당한 카드임을 보이는 것이고, 외부 인증은 터미널이 카드에게 자신이 정당한 카드의 소유자임을 보이는 것이다. 내부 인증과정은 시도-응답(Challenge-response) 방식으로 터미널이 TR(Terminal Random number)를 생성하여 카드에 보내면, 카드는 세션 키인 $K_{ats)}$ 와

CR(Card Random number)을 생성하고, TR을 K_{ats}로 암호화한 암호문 R 값과 CR을 터미널에 전송함으로서 인증과정을 수행한다.

외부 인증은 터미널에서 키 유형, 키의 번호, TR를 보내면 카드는 해당키와 번호를 확인하고 K_{ats}가 아닌 CR과 R 값만 보낸다. 터미널이 K_{ats}를 얻기 위해서는 비밀 키인 K 값을 갖고 있어야 한다. 그림 2.5는 카드와 터미널간 인증 프로토콜에 대한 것이다.

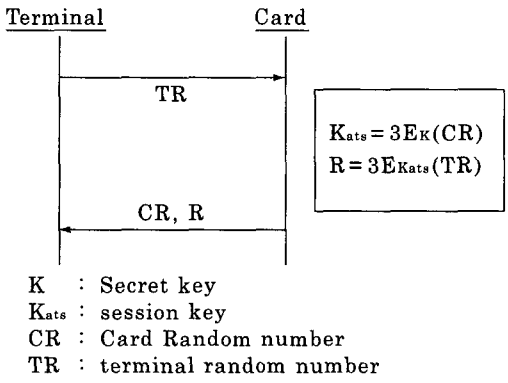


그림 2.5 카드와 터미널간 인증 프로토콜
Fig 2.5 Authentication protocol between card and terminal

2-3. 스마트 카드의 라이프 싸이클

일반적으로 스마트 카드는 카드제조부터 카드가 최종 사용자에게 배포되기까지 4 단계 즉 제조 단계, 카드 초기화 단계, 전용화 단계, 최종 사용자 사용 단계의 라이프 싸이클(Life Cycle)을 갖는다. 제조 단계는 운영체제가 설정되기 전 단계로 카드의 물리적 구성의 완료이다. 이때 제조회사에서는 칩에 고유번호를 저장하므로 카드의 유일성을 보장한다. 카드 초기화(Card Initialization) 단계는 제조된 카드에 파일 시스템을 생성하고 EEPROM 테스트를 통해 응용파일들을 실행할 수 있는 환경을 구성한다. 카드 초기화 작업이 끝나면 카드 발행인에 의해 일반 사용자가 사용할 수 있도록 카드의 발행인 식별 번호와 실행 가능한 코드가 스마트 카드에 저장이 되며 응용에 필요한 파일 구조가 형성된다. 이 과정을 카드의 전용화(Card Personalization)단계라 한다. 이 과정들이 끝나면 최종 사용자가 사용할 수 있는 최종 사용자 단계가 된다. [6][7]

3. 클라이언트-서버 인증 프로토콜의 설계

MPCOS-3DES 스마트 카드를 기반으로 한 클라이언트-서버 인증 시스템을 위해서는 카드에 저장될 자료의 형식을 정의하는 전용화 단계, 인증 과정의 수행 단계로 나뉜다. 인증 과정은 다시 카드와 사용자간의 인증, 클라이언트-서버간 인증, 카드와 서버간의 인증 단계를 거친다.

3-1. MPCOS-3DES 카드의 파일 구조 설계

인증 시스템에서 사용하게 될 카드의 파일

시스템은 MF와 하나의 EF 키 파일로 이루어지는 글로벌 단계와 3개의 DF로 구성된다. 첫 번째 DF는 카드운영 시스템과 제조회사정보, 발행 회사 정보와 카드 정보가 저장되고, 두 번째 DF는 사용자와 카드간 인증에 필요한 파일들이 저장되며 세 번째는 카드와 서버간 인증에 필요한 정보가 저장된다. 구조를 도식화하면 그림 3.1과 같다.

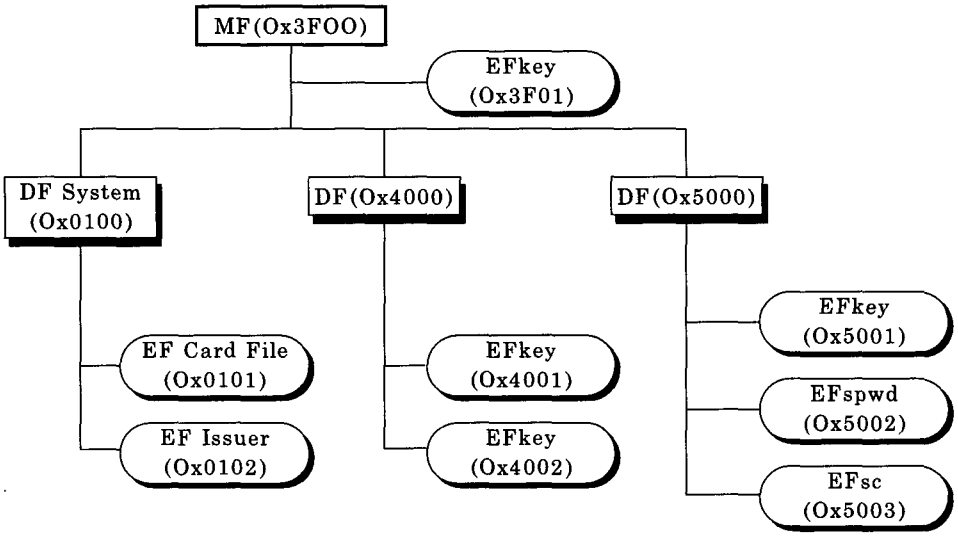


그림 3.1 MPCOS-3DES 카드의 파일 시스템
Fig 3.1 File structure of MPCOS-3DES smart card

3-2. 스마트 카드와 사용자간의 인증

카드와 사용자간의 인증은 사용자가 카드를 판독기에 삽입하고 서버에 접속하려는 프로그램을 실행함으로써 이루어진다. 카드를 삽입한 사용자는 키와 패스워드를 입력하게 되며 입력된 키를 가상키 Kv라하고 크기를 8 자 이상 15 자 이하로 한다. 가상키와 패딩(Padding)을 포함한 16 바이트를 DES로 암호화 하면 16 바이트의 암호문 Kr이 생성되고, Kr을 카드와 터미널간 세션키를 만들기 위한 비밀키로 사용한다. 클라이언트와 스마트 카드간에 세션이 형성되면 패스워드 파일에 저장된 값과 사용자가 입력한 패스워드 *passwd*를 비교함으로서 사용자 인증을 하게 된다. 패스워드는 4바이트의 문자이다.

1. Kv와 *passwd*를 사용자로부터 입력받는다.
2. 클라이언트 - Kr 구하기
   $R = P_n + K_v + PP_n$: 패딩 바이트 개수, Kv : 가상키, P: 패딩 바이트)
   $K_r = E_m(R)$
3. 카드와 터미널간 세션키 형성, *passwd* 얻기
4. 카드로부터 *passwd* 값과 사용자로부터 입력받은 *passwd* 값 비교

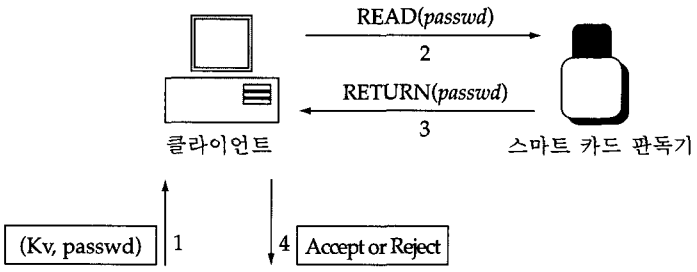


그림 3.2 카드와 사용자간의 인증
Fig 3.2 Authentication between card and user

3-3. 클라이언트와 서버간의 인증

클라이언트는 서버에게 접속 요청을 보내고, 서버는 클라이언트의 IP주소를 이용해 비밀키를 찾는다. 각 IP 마다 서로 다른 키를 사용하기 때문에 서버는 각 클라이언트의 비밀키 Kip를 KeyDB에 저장한다. 사전에 Kip는 안전하게 배포되었다고 가정한다. 서버는 8 바이트의 난수 Kcss를 3중 DES를 사용하여 암호화한 암호문 R을 클라이언트에게 전송한다. 클라이언트는 수신한 암호문을 자신이 갖고 있는 3중 DES 키 Kip로 복호화한 세션키 Kcss로 카드 고유번호를 암호화하여 R'을 서버에게 전송한다. 카드의 고유번호는 시스템 DF내의 EF Card file에 저장되어있다. DES는 동일한 평문에 대해서 동일한 암호문 출력을 생성한다.[1][3] 하지만 클라이언트와 서버간에 매 세션마다 서로 다른 세션키를 이용하여 카드 고유번호를 암호화하기 때문에 매 세션마다 다른 암호문이 전송된다. 서버는 클라이언트로부터 받은 카드 고유번호를 CardDB(각 스마트 카드에 저장된 자료의 데이터베이스)에서 찾아서 카드와 서버간 인증에 필요한 자료를 클라이언트에게 전송한다. 그림 3.3은 위 과정을 도식화한 것이다.

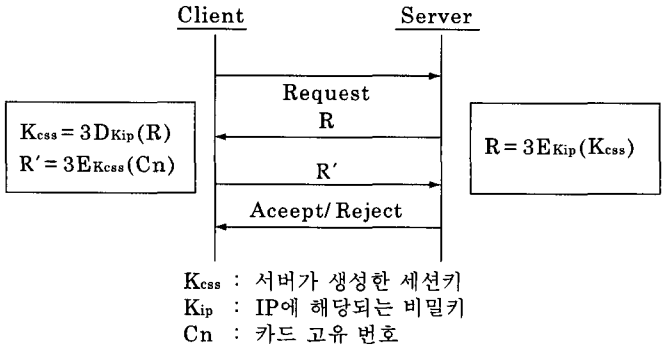


그림 3.3 클라이언트와 서버간의 인증
Fig 3.3 Authentication between client and server

3-4. 서버와 카드간의 인증

클라이언트는 세션키로 암호화 된 카드 고유번호를 전송하고 서버에서는 복호화후 데이터 베이스에서 해당 레코드를 찾는다.

클라이언트/서버간 인증은 세션키 Kcss의 공유과정이다. CardDB에 해당 카드 번호가 없을 경우 서버는 클라이언트를 거부하게되고, 존재한다면 카드내에 서버 접속에 필요한 패스워드 파일의 내용을 얻기 위한 키 EFkey와 비밀코드 파일 EFsc을 클라이언트에게 Kcss로 암호문을 만들어 전송한다. 스마트 카드와 터미널간의 수행은 사용자와 카드간 인증과 동일한 과정을 거치며 단지 비밀코드 부분만 추가되는 것이다. 그림 3.4는 카드와 서버간 인증 과정을 도식화 한 것이다.

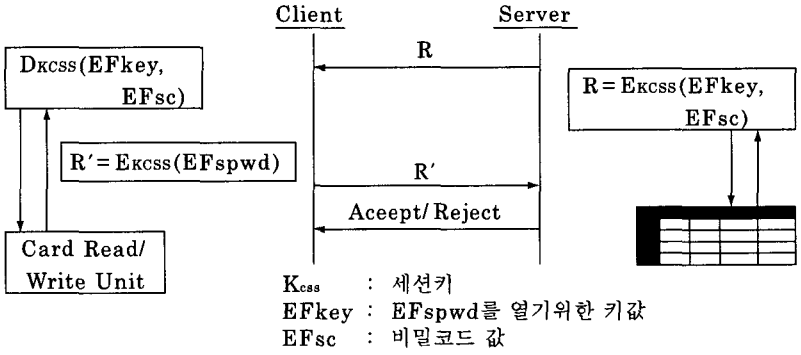


그림 3.4 카드와 서버간의 인증
Fig 3.4 Authentication between card and server

4. 구 현

MPCOS-3DES를 이용한 인증 시스템은 대칭형 암호체계인 DES와 3중 DES를 사용하고 있다. 개발환경은 개인용 컴퓨터에서 구현을 하였으며 운영체제는 WINDOWS 95와 Visual C++ 4.2를 사용하였다. 클라이언트와 서버간의 통신 프로토콜은 TCP/IP를 지원하는 Winsock Version 2.0을 사용했다. 구현은 크게 파일 시스템의 생성 즉 카드의 전용화 단계와 클라이언트 서버 환경의 구축이다. 대칭형 암호인 DES는 블록 암호체계이므로 일반 스트림을 암호화할 경우 1 바이트의 패딩 바이트 길이와 Data, 0x00의 패딩 바이트를 추가하여 구성한다

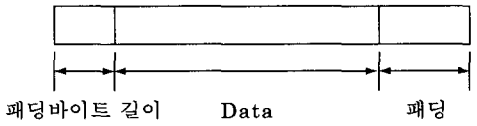


그림 4.1 스트림의 블록 변환
Fig 4.1 Blocking of stream

다음 표는 MPCOS-3DES 카드를 이용하여 인증 시스템을 구현하기 위해 카드의 전용화 작업을 마친 후 파일 시스템을 표로 작성한 것이다.

파일형	ID	크기	암호방식	제어조건	내 용
MF	0x3F00	0	DES		없음
EF key	0x3F01	24		읽기불가	글로벌 키값
DF system	0x0100	6	DES	EF key을 통한 파일의 생성	"SYSTEM"
EFCard file	0x0101	12		읽기만 허용	카드 제조 번호
EF Issuer file	0x0102	12		읽기 허용	발행인 자료
사용자와 카드간 인증					
DF	0x4000	8	3중 DES	읽기, 쓰기, 수정은 보안 메시징 사용	"Personal"
EFkey	0x40001	24		읽기 불가	3중 DES 키
EF pwd	0x4002	4		읽기, 쓰기, 수정은 보안 메시징 사용	사용자가 정의한 패스워드
카드와 서버간 인증					
DF	0x5000	8	3중 DES	보안 메시징 사용	"server"
EFkey	0x5001	24		읽기 불가	3중 DES 키
EFspwd	0x5002	4		보안메시징 사용	패스워드
EFsc	0x5003	8		읽기 불가	비밀코드

표 4.1 카드 파일 시스템
Table 4.1 File system of smart card

전체 인증 과정을 정리하면 다음과 같이 3단계의 과정을 거친다.

Step 1: 카드와 사용자간의 인증

1. 사용자가 Kv(8~15자)와 Passwd(4자)를 입력
2. 카드 판독기와 클라이언트간 세션키 공유 후 카드의 패스워드 요청
3. EF_{pwd}를 터미널에 보냄.
4. **If** (Passwd $==EF_{pwd}$)
   Then next step.
   else Reject

Step 2: 클라이언트와 서버간 인증

5. 클라이언트가 서버에 접속요청
6,7. 서버는 클라이언트 IP와 매핑되는 3중 DES 키 Kip를 찾고, 세션 키 생성
8. R_1전송
   $R_1 = 3E_{Kcss}(Cn)$
   3E : 3중 DES 암호화함수
   Kip : 해당 IP에 매핑되는 3중 DES 키 값
   K_{CSS} : 서버에서 생성한 세션키
9. 클라이언트는 카드번호를 암호화시킨 R' 전송
   $R' = 3E_{Kcss}(Cn)$
   Cn : Card serial Number

Step 3: 서버와 카드간의 인증

10.11. 서버는 cardDB에서 클라이언트의 카드번호 자료를 탐색 (클라이언트가 서버에게 보낼 패스워드를 얻을 수 있는 키와 비밀코드)
12. 카드의 파일 접속에 필요한 자료를 암호화 후 전송
13. 서버접속에 필요한 패스워드를 서버에 전송
14. Accept/Reject 결과를 클라이언트에게 전송.

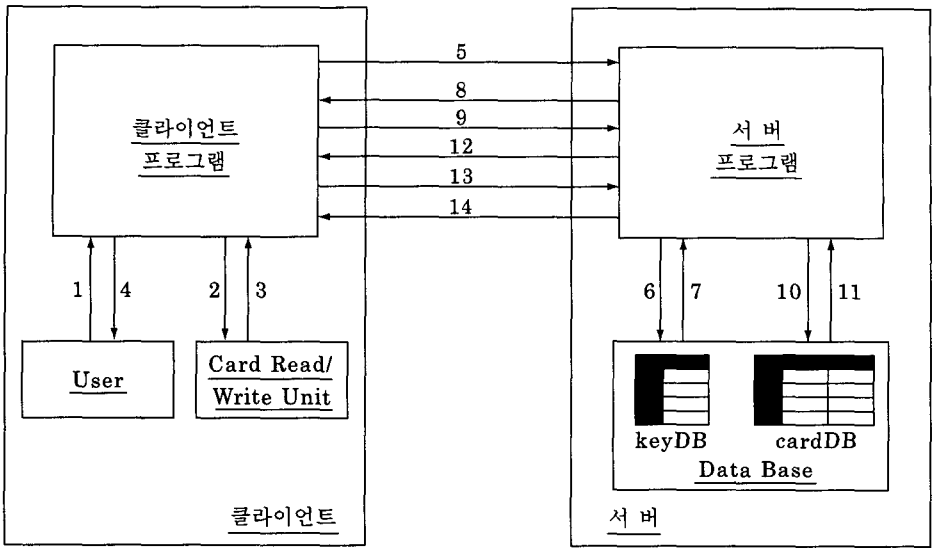


그림 4.2 동작 시나리오
Fig 4.2 Operating flow

위 과정을 수행한 것이 그림 4.3 과 4.4 이다. 클라이언트 프로그램에서 보듯이 3단계로 이루어 지며 각 단계마다 클라이언트와 카드, 클라이언트와 서버, 클라이언트를 통한 카드와 서버간의 인증 과정에 필요한 자료의 교환을 볼 수 있다.

그림 4.3 클라이언트 프로그램
Fig 4.3 Client program

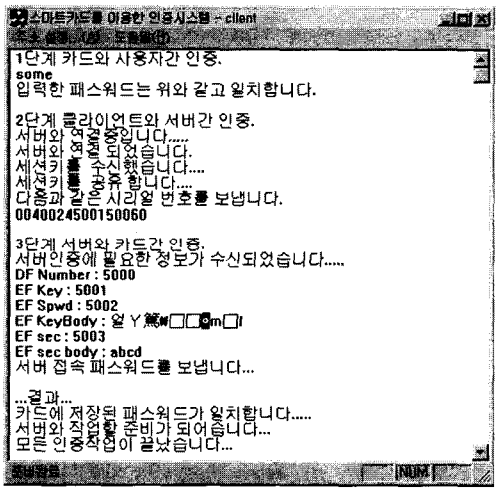


그림 4.4 서버 프로그램
Fig 4.4 Server program

5. 결 론

본 논문에서는 MPCOS-3DES 카드를 이용한 클라이언트 서버간 인증 시스템을 구현하였다. 카드의 안전한 보안 메카니즘을 통해 패스워드 혹은 암호화 키의 저장을 통해 사용자는 단지 카드와의 인증 과정만 수행하면 서로 다른 시스템에 접속할 수 있다. 스마트 카드를 이용한 인증 시스템을 사용자와 카드, 클라이언트와 서버 그리고 카드와 서버간의 3단계 인증과정을 구현하였다. 현재는 대칭형 암호시스템을 사용하는 카드 시스템을 사용하고 있지만 비대칭형 암호 시스템을 적용할 수 있는 카드를 사용할 경우 디지탈 서명이나 키의 관리에서 보다 다양한 보안 메카니즘이 제공될 수 있을 것이다.

참고 문헌

[1] S. Garfinkel & G. Spafford, Practical Unix & Internet Security, O'REILLY, 1996. 4

[2] S. Garfinkel & G. Spafford, Web Security & Commerce, O'REILLY, 1996. 6

[3] Charlie Kaufman, Radia Perlman and Mike Speciner, Network Security Private Communication in a Public World, Prentice Hall,, 1995

[4] D. Naccache and D M'Raihi, Cryptographic Smart Card, IEEE Micro, vol. 1, No. 6, 1996, pp. 31-34

[5] C. Neuman, Security, Payment, and Privacy for Network Commerce, IEEE Journal on Selected Areas of Communications, vol.13, no.8 , 1995, pp 1523-1531.

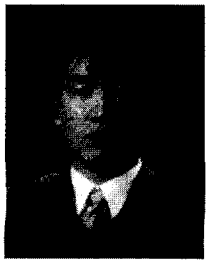
[6] José Luis Zoreda and José Manuel Otón, Smart Cards, Artech House Publishers, 1994

[7] MPCOS Refence Manual ver 1.0, GEMPLUS, 1996.3

[8] MPCOS-3DES Refernce Manual ver 1.0, GEMPLUS, 1996.3

[9] MPCOS-3DES Interface Library ver 1.0, GEMPLUS, 1996.3

□ 著者紹介



김 경 훈

1996년 2월 단국대학교 전자계산학과(이학사)
1998년 2월 단국대학교 전자계산학과(이학석사)
1999년 9월 ~ 현재 고려대학교 전산학과 박사과정

※ 주관심분야 : 암호이론, 전자상거래

박 창 섭

1983년 연세대학교 경제학 학사
1983년 한국 IBM System Administration 근무
1987년 LEHIGH Univ 전자계산학 석사
1990년 LEHIGH Univ 전자계산학 박사
1990년 단국대학교 전자계산학과 조교수
1994년 ~ 현재 단국대학교 전자계산학과 부교수

※ 주관심분야 : 부호이론, 암호학