사용자 로그 추적이 가능한 이동식 저장매체 보안 솔루션 설계 및 구현

김현진, 고성종, 이선호, 이임영
순천향대학교 컴퓨터소프트웨어공학과
e-mail:[teeniebears, lunatics, sunho431, imylee]@sch.ac.kr

Design and implementation of traceable secure solution for removable storage medium

Hyun-Jin Kim, Sung-Jong Go, Sun-Ho Lee, Im-Yeong Lee
Department of Computer Software Engineering, Soonchunhyang University

요 약

IT 기술이 점점 발전함에 따라 이동식 저장매체는 가격대비 고용량 및 높은 휴대성을 제공하고 다양한 방식으로 접근하여 많은 사용자 계층을 확보하고 있다. 이러한 이동식 저장매체의 분실 및 도난을 통하여 각종 개인 정보 유출 사건사고와 같은 2차 피해가 빈번하게 발생됨에 따라 보안 솔루션이 개발되어 사용자들에게 제공되고 있다. 하지만 기존에 개발되어 제공되는 이동식 저장매체 보안 솔루션은 보안 취약점이 발견되고, 부족한 사용자 편의성으로 인해 더욱 안전하고 사용자 편의를 제공하는 보안 솔루션의 필요성이 증가되고 있다. 따라서 본 논문에서는 기존의 이동식 저장매체 보안 솔루션과 차별화되어 2차 피해 발생을 예방하고, 더욱 안전한 보안성 및 사용자 편의성을 제공하는 이동식 저장매체 보안 솔루션을 설계하고 구현한다.

1. 서론

IT기술력의 고도화로 전 세계는 초고속 광 통신망으로 연결되어 고속의 네트워크를 통해 고용량의 데이터를 주고받을 수 있게 되었다. 그에 따라 해당 데이터를 저장하기 위한 방법으로 고용량의 저장매체가 개발 및 생산되어 보급되었다. 데이터의 휴대성을 위한 이동식 저장매체의 경우 USB(Universal Serial Bus)메모리의 등장으로 환경이 급변하게 되었다.

USB Flash Driver 혹은 USB Disk 등으로 불리는 USB 메모리는 가벼운 무게와 작은 크기로 인해 높은 휴대성을 제공하며, 고용량의 제품들이 저렴한 가격으로 보급되고 있어 많은 사용자들로 하여금 각광받고 있다. 그에 따라 높은 휴대성을 제공한 만큼 이동식 저장매체의 분실 및 도난이 빈번해졌으며 개인정보 및 기업의 주요 정보 유출 사고가 발생하는 문제점이 발생되었다. 이러한 정보 유출 사고는 보안 분야에 있어서 지속적으로 심각성이 대두되고 있다[1].

이동식 저장매체에 대한 보안 중요성이 증가됨에 따라 업계에서는 각자의 특징을 내세워 정보 유출 사고를 방지하기 위한 방법으로 보안영역을 제공하는 보안 솔루션을 개발하여 출시하기 시작했다. 하지만 기존에 출시된 보안 솔루션 제품의 경우 악의적인 사용자가 사용자 인증을 시도한 경우 이를 추적하는 기술이 제공되지 않거나 오프라인 환경에서 이루어지는 사용자 인증시도에 대한 미 대응으로 인하여 2차 피해가 발생되는 문제점을 가지고 있다[2].

따라서 본 논문에서는 위에 서술한 문제점을 해결하기 위하여 파일시스템 구조를 분석하고 이를 이용하여 안전한 사용자 인증 및 보안영역 제공 메커니즘을 구현한다. 또한 오프라인 환경에서 발생되는 사용자 인증 시도를 차단하고, 온라인 환경에서는 인증을 시도한 PC의 접속정보를 보관하여 사후추적이 가능하게 함으로써 2차 피해 문제를 감소시키고 보안 취약점을 보완한다.

본 논문의 구성은 2장에서 파일시스템 구조 분석에 대해 기술하고 3장에서는 보안 솔루션의 보안요구사항을 기술한다. 4장에서는 사용자 로그 추적이 가능한 이동식 저장매체 보안 솔루션 설계를 제안하고, 5장에서는 제안한 보안 솔루션 설계에 대한 구현을 기술한다. 마지막으로 6장에서는 결론을 맺는다.

2. 관련 연구

본 장에서는 안전한 보안영역 제공 메커니즘을 위한 파일 시스템 구조 분석에 대하여 알아본다.

2.1. MBR(Master Boot Record)

MBR은 저장매체에 저장된 정보들이 어떻게 위치하고

있는지를 식별하여 컴퓨터의 주기억장치에 적재할 수 있도록 하기 위한 정보이다. 물리 디스크의 0번 섹터에 위치하며 부팅에 필요한 프로그램인 Boot Code와 파티션 테이블을 포함하고 있다.

각 섹터는 512Byte로 구성되어 있으며 MBR의 경우 MBR임을 나타내기 위해 섹터의 가장 마지막 2Byte를 매직코드 55AA를 위치시킨다. 매직 코드 앞으로는 각 파티션의 정보를 타나내는 16Byte 길이의 파티션 테이블이 4개가 존재한다. Boot Code는 가장 앞부분의 446Byte 길이를 가지며, 파티션 데이터가 정상적이지 않거나 부팅 가능한 파티션이 없는 상황 등의 예외처리를 처리한다. 또한 최종적으로 부팅 가능한 파티션의 실제 주소를 계산하여 해당 파티션의 Boot Record를 호출한다[4].

2.2 PBR(Partition Boot Record)

파티션 테이블의 LBA Begin이 가리키는 곳에 해당 파티션의 정보를 나타내는 PBR이 존재하게 되며 이는 파티션의 시작 섹터 크기, 이름, 포맷의 종류 등 파티션의 정보를 저장하고 있다.

이러한 PBR이 저장되는 위치는 해당 볼륨의 첫 번째 섹터이며, 만약 디스크에 볼륨이 여러 개 있다면 각각의 볼륨에 Boot Record도 하나씩 존재하게 된다. 그러므로 파티션 생성을 위해서는 PBR 정보의 정확한 입력이 필요하다[4].

3. 보안 요구사항

본 장에서는 사용자 로그 추적 서비스를 제공하는 이동식 저장매체 보안 솔루션에 대한 보안 요구사항을 알아본다.

- 사용자 인증(Authentication)

인증되지 않은 사용자가 저장매체의 보안영역에 접근 및 서비스 사용을 차단해야 하며, 그 신원이 거짓이 아닌 정당한 사용자라는 것을 검증할 수 있어야 한다. 또한 사용자 인증과정의 우회가 불가능해야 한다.

- 접근 제어(Access Control)

이동식 저장매체 보안영역의 정보 자원에 대한 읽기나 변경 등의 모든 행위에 대해 그 권한을 명백히 구분하여 허가되지 않은 접근 시도를 사전에 차단할 수 있도록 하는 통제가 필요하다.

- 기밀성(Confidentiality)

클라이언트와 서버간의 통신은 정당한 객체만이 확인할 수 있어야 한다. 또한 서버에 저장되는 클라이언트의 접속 정보는 비윤리적인 서버관리자에게 노출되지 않도록 하는 기밀성이 제공되어야 한다.

- 무결성(Integrity)

사용자 인증 값 및 암호화 등에 사용되는 키 값은 위·변조되거나 파괴되지 않도록 해야 한다. 만약 위조, 삭제 및 변화가 되었다면 그 사실을 확인할 수 있어야 한다.

- 가용성(Availability)

사용자에게 실시간으로 지속적인 로그 추적 서비스를 제공하기 위해 사용자 인증은 온라인 환경에서만 이루어지도록 해야 한다.

4. 제안방식

본 논문에서 제안하는 방식은 이동식 저장매체에 대해 사용자가 원하는 크기의 안전한 보안 영역을 설정하여 제공하며, 사용자 인증 시 온라인 환경에서만 이루어진 후 보안영역 접근이 가능하도록 한다. 또한 사용자 인증을 시도한 PC의 접속 정보를 서버에 전송하여 기록함으로써 실시간 사용자 추적 혹은 사후 관리를 할 수 있도록 한다.

4.1 시스템계수

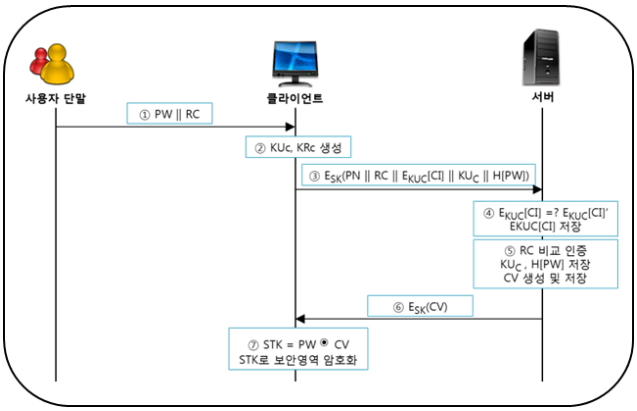
본 제안방식에서 사용되는 시스템계수는 다음과 같다.

- * : 참여 객체(C: Client, S: Server)
- KR_* : *의 개인키
- KU_* : *의 공개키
- PW : 사용자 인증 비밀번호
- NPW : 변경할 사용자 인증 비밀번호
- RC : 솔루션 구입 시 발급된 구매자 인증을 위한 등록 코드
- SK : 안전하게 공유된 세션키
- PN : 이동식 저장매체의 기기식별번호
- CI : 클라이언트의 접속 정보
- CV : 보안영역 암호키를 생성하기 위해 필요한 사용자 인증 비밀번호에 대응하는 키 생성자
- STK : 보안영역 암호키

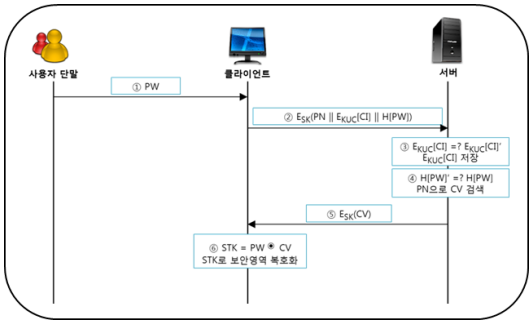
4.3 보안영역 설정 단계

해당 보안 솔루션을 구입한 사용자는 보안영역 설정 및 사후 사용자 로그 추적 서비스를 위해 필요한 설정을 다음과 같이 진행한다(그림 1).

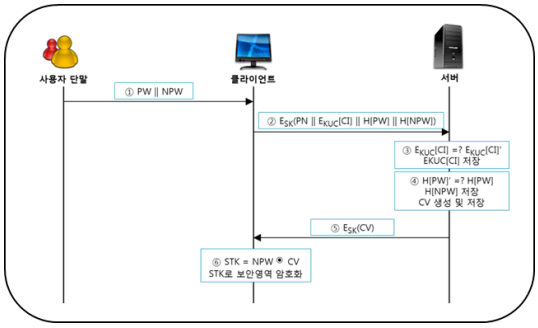
Step 1. 사용자는 사용자 인증을 위해 필요한 비밀번호 PW와 솔루션 구입 시 발급된 구매자 인증 코드 RC를 클라이언트에 입력한다.



(그림 1) 보안영역 설정 단계



(그림 2) 사용자 기기 인증 단계



(그림 3) 비밀번호 갱신 단계

Step 2. 클라이언트는 클라이언트의 개인키 및 공개키 쌍을 생성한다.

Step 3. 클라이언트는 서버로부터 안전하게 전송된 세션키 SK로 아래 내용을 암호화하여 서버에 전송한다.

$$E_{SK}[PN \| RC \| E_{KU_C}[CI] \| KU_C \| H[PW]]$$

Step 4. 서버는 수신된 PN과 RC로 사용자를 인증하며, 인증에 성공하면 KU_C와 $H[PW]$를 서버의 DB에 저장한다.

Step 5. 서버는 통신패킷으로부터 획득한 실제 클라이언트의 접속정보 CI를 클라이언트 공개키로 암호화한 값 $E_{KU_C}[CI]$과 클라이언트로부터 전송된 $E_{KU_C}[CI]'$을 비교하여 접속정보 진위 여부를 확인하고 $E_{KU_C}[CI]$를 저장한다.

Step 6. 서버는 보안영역 암호화키를 만드는데 필요한 CV를 생성하고, 이를 세션키로 암호화하여 클라이언트에 전송한다.

Step 7. 클라이언트는 PW와 CV를 XOR하여 실제 보안영역 정보를 암호화하는 키 STK를 생성하고, 보안영역의 정보를 암호화한다.

4.4 사용자 기기 인증 단계

보안영역에 접근하기 위해선 사용자 인증이 필요하며 해당과정은 다음과 같이 진행된다(그림 2).

Step 1. 사용자는 인증을 위해 필요한 비밀번호 PW를 클라이언트에 입력한다.

Step 2. 클라이언트는 서버로부터 안전하게 전송된 세션키 SK로 아래 내용을 암호화 하여 서버에 전송한다.

$$E_{SK}[PN \| E_{KU_C}[CI] \| H[PW]]$$

Step 3. 서버는 수신된 PN과 $H[PW]$로 사용자를 인증하며, 인증에 성공하면 PN에 해당하는 CV값을 검색한다.

Step 4. 서버는 통신패킷으로부터 획득한 실제 클라이언트의 접속정보 CI를 클라이언트 공개키로 암호화한 값 $E_{KU_C}[CI]$과 클라이언트로부터 전송된 $E_{KU_C}[CI]'$을 비교하여 접속정보 진위 여부를 확인하고 $E_{KU_C}[CI]$를 저장한다.

Step 5. 서버는 검색된 CV를 세션키로 암호화하여 클라이언트에게 전송한다.

Step 6. 클라이언트는 PW와 CV를 XOR하여 실제 보안영역 정보를 복호화하는 키 STK를 생성하고, 보안영역의 정보를 복호화하여 보안영역에 접근한다.

4.5 비밀번호 갱신 단계

사용자가 인증을 위한 비밀번호를 변경하고자 할 경우 다음과 같은 과정을 통하여 비밀번호가 생성된다(그림 3).

Step 1. 사용자는 인증 위해 필요한 비밀번호 PW와 새롭게 변경할 비밀번호 NPW를 클라이언트에 입력한다.

Step 2. 클라이언트는 서버로부터 안전하게 전송된 세션키 SK로 아래 내용을 암호화 하여 서버에 전송한다.

$$E_{SK}[PN \| E_{KU_C}[CI] \| H[PW] \| H[NPW]]$$

Step 3. 서버는 수신된 PN과 $H[PW]$로 사용자를 인증하며, 인증에 성공하면 $H[NPW]$를 서버의 DB에 저장한다.

Step 4. 서버는 통신패킷으로부터 획득한 실제 클라이언트의 접속정보 CI를 클라이언트 공개키로 암호화한 값 $E_{KU_C}[CI]$과 클라이언트로부터 전송된 $E_{KU_C}[CI]'$을 비교하여 접속정보 진위 여부를 확인하고 $E_{KU_C}[CI]$를 저장한다.

Step 5. 서버는 보안영역 암호화키를 만드는데 필요한 CV를 재생성하고, 이를 세션키로 암호화하여 클라이언트에게 전송한다.

Step 6. 클라이언트는 NPW와 CV를 XOR하여 실제 보안영역 정보를 암호화하는 키 STK를 재생성하고, 보안영역의 정보를 암호화한다.

4.6 사용자 추적 단계

이동식 저장매체의 분실 시 혹은 사용자 접속 정보를 추적하고자 할 경우 다음과 같은 과정을 수행한다(그림 4).

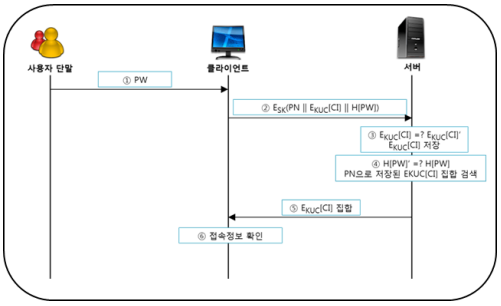
Step 1. 사용자는 인증을 위해 필요한 비밀번호 PW를 클라이언트에 입력한다.

Step 2. 클라이언트는 서버로부터 안전하게 전송된 세션키 SK로 아래 내용을 암호화 하여 서버에 전송한다.

$$E_{SK}[PN \| E_{KU_C}[CI] \| H[PW]]$$

Step 3. 서버는 수신된 PN과 $H[PW]$로 사용자를 인증한다.

Step 4. 서버는 통신패킷으로부터 획득한 실제 클라이언트의 접속정보 CI를 클라이언트 공개키로 암호화한 값



(그림 4) 사용자 추적 단계

$E_{KU_C}[CI]$과 클라이언트로부터 전송된 $E_{KU_C}[CI]'$을 비교하여 접속정보 진위 여부를 확인하고 $E_{KU_C}[CI]$를 저장한다.

Step 5. 서버는 *PN*으로 사전에 저장된 암호화된 접속정보들을 검색하고 이를 세션키로 암호화하여 클라이언트에게 전송한다.

Step 6. 클라이언트는 자신의 공개키로 암호화 저장된 클라이언트들의 접속 정보들을 자신만이 알고 있는 개인키로 복호화하여 접속 정보를 추적한다.

5. 제안방식 구현

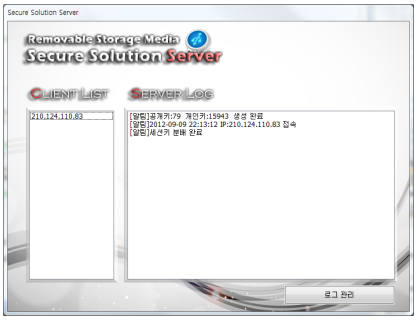
4장에서 설계한 내용을 기반으로 사용자 로그 추적이 가능한 이동식 저장매체 보안 솔루션을 구현하였다[5].

5.1 보안 솔루션 클라이언트 구현

보안 솔루션 클라이언트는 정상적인 인증 완료 후 사용자가 이용할 수 있는 서비스 버튼을 위치시켜 간단하게 동작할 수 있도록 구성하였다. 사용자는 보안영역 설정,

(그림 5) 보안 영역 할당 화면

(그림 6) 클라이언트 초기 화면



(그림 7) 사용자 추적 서버

드라이브 선택, 비밀번호 변경, 사용자 추적 서비스를 이용할 수 있다. 각 서비스 버튼 클릭 시 관련 화면을 활성화하여 서비스를 이용한다(그림 5, 6).

5.2 사용자 추적 서버 구현

사용자 추적 서버는 사용자 인증을 시도한 PC의 접속정보를 DB에 저장함으로써 추후 이동식 저장매체 분실 혹은 실시간 사용자 추적 서비스 이용 시에 사후 관리가 가능하도록 한다. 뿐만 아니라 사용자 인증 및 보안영역 암호키 생성에 필요한 데이터를 통신한다.

서버의 전체적인 구성은 현재 접속 중인 클라이언트의 IP주소와 실시간 서버 로그 기록을 보여주며 서버 관리자의 편의성을 위한 로그 관리 기능 버튼을 배치시켰다(그림 7).

6. 결론

이동식 저장매체의 높은 휴대성으로 인한 분실 및 도난으로 저장매체에 저장된 사용자의 개인정보 및 사내 주요 정보들이 노출되는 사건이 급증함에 따라 보안 솔루션에 대한 중요성이 증가되었다. 뿐만 아니라 분실 및 도난 사건 발생 시 사후 관리를 통한 2차 피해 방지의 필요성이 증가되었다. 이에 본 논문은 인증된 사용자만이 접근할 수 있는 보안영역 제공 기술을 구현하였으며, 사용자 인증 시 PC의 접속정보를 저장하여 사후 추적이 가능하도록 구현하였다.

향후 비밀번호 분실 관련 추가적인 보안 솔루션 서비스를 추가 개발한다면 더욱 편리하고 안전한 이동식 저장매체 사용 환경을 이루어 낼 것으로 본다.

참고문헌

[1] 이선호, 이임영, “USB 메모리를 위한 보안 솔루션에 관한 연구”, 멀티미디어학회 논문지, 13(1), 2010.1.

[2] 이선호, 이임영, “보안 USB 동향에 관한 고찰”, 정보처리학회, 17(2), 2010.

[3] William Stallings, “컴퓨터 보안과 암호”, 그린출판사, 2011.

[4] 정준석, 정원용, “임베디드 개발자를 위한 파일시스템의 원리와 실습”, 한빛미디어, 2007.

[5] 성윤정, “Visual C++ MFC 윈도우 프로그래밍”, 인피니티북스, 2009.