情報保護學會論文誌
第18卷 第4號, 2008. 8

Trusted Mobile Platform 환경에서의 안전한 비밀 데이터 유지(이전) 방안*

강 동 완[1†], 이 임 영[1‡], 한 진 희[2], 전 성 익[2]
[1]순천향대학교 컴퓨터학부, [2]한국전자통신연구원

A Secure Maintenance Scheme of Secret Data on Trusted Mobile Platform Environment*

Dong-Wan Kang[1†], Im-Yeong Lee[1‡], Jin-Hee Han[2], Sung-Ik Jun[2]
[1]Department of Computer Science, Soonchunhyang University,
[2]Electronics and Telecommunications Research Institute

요 약

현대 사회는 정보화 사회로써 가치있는 많은 정보들이 온라인상에서 송수신되고 있다. 특히, 무선 통신을 기반으로 하는 모바일 환경은 유선 통신에 비해 유연한 특징을 가지고 급속도로 발전하고 있지만, 모바일 환경의 발전과 함께 민감한 정보들이 온라인상에 노출되면서 보안에 대한 요구가 증가하게 되었다. 따라서 신뢰 컴퓨팅을 표준화 하고 있는 TCG(Trusted Computing Group)는 모바일 환경의 보안을 위해 하드웨어 기반의 보안 모듈인 MTM(Mobile Trusted Module)을 제안하였다. MTM은 플랫폼에 임베디드되어 사용자 프라이버시와 플랫폼 무결성을 보호하고 물리적으로 안전하지만 강한 보안 기능을 제공하는 만큼 비밀 데이터를 다른 곳으로 이전할 때 보안적인 접근이 요구된다. 본 논문에서는 TCG의 표준 및 기존에 연구된 비밀 데이터 이전 방안에 대해 분석하고, 사용자 인증모듈인 USIM(Universal Subscriber Identity Module)을 사용한 안전한 비밀 데이터 유지 방안을 제안한다.

ABSTRACT

Modern society as an information society, a lot of information is communicated in on-line. Specially, mobile environment based on radio communication has a characteristic of flexibility compared with wire communication and is developed rapidly. However, the more mobile technology is developed the more security for sensitive information is needed. Therefore, MTM(Mobile Trusted Module) is developed and promoted by TCG(Trusted Computing Group), which is an industry standard body to enhance the security level in the mobile computing environment. MTM, hardware security module for mobile environment, offers user's privacy protection, platform integrity verification, and individual platform attestation. On the other hand, secure migration scheme is required in case secret data or key is transferred from one platform to the other platform. In this paper, we analyze migration schemes which were described in TCG standard and other papers and then propose security maintenance scheme for secret data using USIM(Universal Subscriber Identity Module).

Keywords : Mobile Trusted Module, Migration, Maintenance

1. 서 론

현대 무선 통신의 발전은 기존의 통신 환경을 크게 변화시키게 되었다. 1980년대부터 AMPS(Advanced Mobile Phone System)를 시작으로 하여 1990년대 TDMA(Time Division Multiple Access), CDMA(Code Division Multiple Access)를 기점으로 빠르게 성장한 무선 통신 기술은 기존의 음성통신만 가능했던 무선 통신을 멀티미디어를 포함하는 대용량의 데이터를 처리할 수 있을 만큼 발전시켰다. 기존의 무선 단말기들이 음성 및 간단한 문자 메시지만 가능했던 때와 달리 이제는 인터넷과 영상통화, 그리고 카메라 기능까지 통합하는 스마트 단말기로 바뀌고 있다. 그러나 단말기들의 기능이 다양화되고 여러 무선 통신 능력을 갖추게 됨에 따라 소프트웨어가 복잡해지고 그에 상응하는 소프트웨어 취약점과 하드웨어 공격에 따른 보안 취약점 또한 증가하게 되었다[2,3,7]. 모바일 악성코드 뿐 아니라 단말기 분실에 따른 제 3자의 하드웨어적인 접근에 있어 기존의 소프트웨어 보안은 상대적으로 취약할 수밖에 없다[3,4,15,16]. 따라서 하드웨어 보안 모듈을 사용하여 기존의 소프트웨어 보안의 취약점을 해결하고 모든 산업계에서 사용할 수 있는 개방형 보안 플랫폼을 제시하기 위한 신뢰 컴퓨팅 그룹(TCG : Trusted Computing Group)이 2003년 Intel, IBM, AMD 등의 주요 IT 기업들에 의해 결성되었다[10,12]. 본 논문에서는 TCG가 제안한 모바일 환경에 적합한 보안 플랫폼인 TMP(Trusted Mobile Platform)에서 플랫폼 변경에 따른 비밀 데이터 이전 방안에 대한 문제점을 지적하고자 한다. 플랫폼에 임베디드되는 보안 모듈로써의 MTM은 강한 보안성을 가지지만 그 만큼 암호키를 외부에 노출시키지 않기 때문에 보호된 데이터의 안전한 이동 방안이 필요하다. 따라서 본 연구에서는 향후 사용자 인증 모듈로 널리 사용될 USIM과 단말기 검증과 모바일 환경에 적합한 보안 모듈인 MTM을 사용하여 안전한 데이터 이전 방안을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 TCG에 대한 간략한 내용과 MTM을 적용하는데 따른 보안 요구사항과 모바일 환경의 보안 요구사항을 살펴본다. 이어 3장에서는 기존에 연구된 비밀 데이터 이전 방안으로써 TCG의 migration과 maintenance, TMA(TPM Migration Authority)를 사용한 이전 방안에 대해 분석하고 4장에서는 USIM(Universal Subscriber Identity Module)을 사용한 제안방식을 설명한다. 5장에서 2장에서의 보안 요구사항을 기반으로 제안방식을 분석하고 마지막으로 6장에서 결론과 향후 연구방향을 제시한다.

접수일 : 2008년 3월 27일; 수정일 : 2008년 5월 18일;
채택일 : 2008년 6월 19일
* 본 연구는 정보통신부 및 정보통신연구진흥원의 IT신성장 동력핵심기술개발사업의 일환으로 수행하였음. [2006-S-041-02, 차세대 모바일 단말기의 보안 및 신뢰 서비스를 위한 공통보안 핵심 모듈 개발]
† 주저자, lupin428@sch.ac.kr
‡ 교신저자, imylee@sch.ac.kr

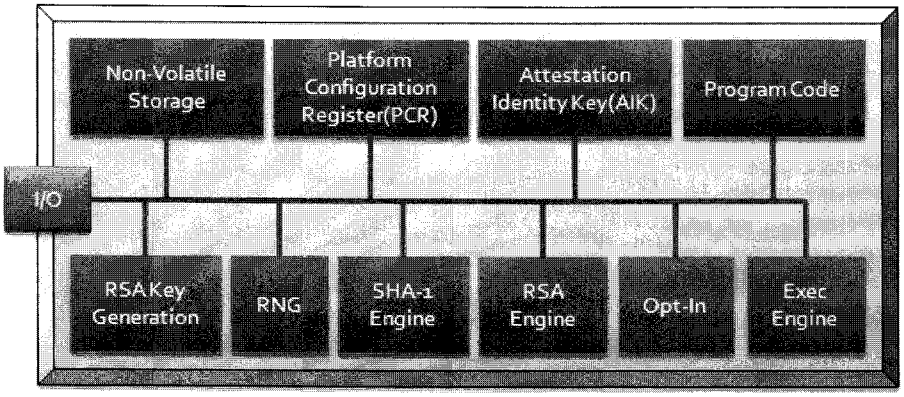
Ⅱ. 연구배경

본 장에서는 신뢰 컴퓨팅에 대한 표준화 단체인 TCG의 보안기술에 대해 살펴보고, 모바일 환경에서 MTM 적용과 관련된 제약사항과 그에 따른 보안 요구사항을 분석한다.

2.1 TCG와 MTM 사용의 제약사항

TCG는 신뢰컴퓨팅을 위한 보안기술로써 하드웨어 기반의 보안 모듈을 표준화하고 있는데 일반 PC(Personal Computer) 환경을 위한 TPM(Trusted Platform Module)과 TPM을 모바일 환경에 적용하기 위한 MTM(Mobile Trusted Module)이 각각 2004과 2006년에 제안되었으며 현재까지 지속적으로 표준화가 진행 중이다. TPM은 보안기능을 제공하기 위해 [그림 1]과 같은 하드웨어 구조를 가지고 있으며, TPM의 구성요소 중 키와 관련된 몇 가지 주요한 항목을 살펴보면 다음과 같다[11,17,18].

- EK(Endorsement Key) : EK는 TPM의 유일한 키로써 TPM 제조 시 제조사가 생성하여 TPM 내부의 안전한 비휘발성 저장소에 저장하는 개인키/공개키 쌍이다. EK의 개인키는 TPM 내부에서 외부로 노출이 되지 않고, 변경되지 않으며 제조사의 개인키에 의해 서명된다.
- SRK(Storage Root Key) : SRK는 TPM이 안전한 저장소를 제공하기 위해 사용되는 핵심적인 키이다. 안전한 저장소는 일반적인 물리 저장 공간 중에 SRK로 보호된 영역을 의미하며 이 공간에 저장되는 데이터는 SRK에 의해서 암호화되어 저장되기 때문에 정당한 절차를 거치지 않으면 해당 데이터를 알 수 없다.



〔그림 1〕 TPM의 하드웨어 구성

- AIK(Attestation Identity Key) : TPM에서는 AIK를 만들어 각각의 용도에 따른 식별키를 가지게 된다. 사용자는 AIK를 여러 개 만들 수 있으며 이 AIK는 인증기관인 CA(Certificate Authority)로부터 정당한 AIK임을 인증 받는다.
- PCR(Platform Configuration Register) : TPM은 시스템 부팅시 BIOS(Basic Input Output System)보다 먼저 동작하여 BIOS부터 차례로 시스템을 점검해 나간다. 이러한 secure booting 과정을 위해 TPM은 플랫폼의 시스템 소프트웨어와 운영체제 및 응용 계층의 특정 애플리케이션에 이르기까지 플랫폼의 무결성 정보를 모두 저장하고 관리하며 수집된 무결성 정보를 해쉬하여 TPM 내부의 휘발성 저장소인 PCR의 해당 인덱스에 저장한다.

하지만 MTM 역시 제약사항이 존재한다. MTM은 플랫폼에 임베디드되어 사용되는 보안 모듈로써 사용자에게 강한 안전성을 제공하지만[5,13,14], 이 안전성은 해당 플랫폼을 벗어나 다른 플랫폼으로의 데이터 이전에 있어 큰 제약사항이 된다. MTM에서의 키 관리는 모듈 내부 안전한 저장소의 SRK를 기반으로 이루어지는데 SRK의 개인키는 외부로 유출이 되지 않는다. 따라서 SRK에 의해 보호되고 있던 데이터들을 다른 플랫폼으로 이전하기 위한 방안이 필요하다. 여기서의 이전 대상인 비밀 데이터는 SRK, 혹은 사용자의 비밀 키 및 일반 응용 프로그램에서 사용되는 암호 키 등을 대상으로 한다..

2.3 보안 요구사항

모바일 단말기는 사용자가 오랜 시간 휴대하고 다니기 때문에 사용자의 프라이버시와 관련된 정보를 많이 저장하고 있다[3,6]. 따라서 모바일 단말기를 사용함에 있어 보안 요구사항을 아래와 같이 정의할 수 있다.

- 사용자 인증 : 모바일 단말기를 사용하고자 하는 사용자가 단말기의 정당한 소유자인지 검증할 수 있어야 한다. 정당한 소유자가 아니면 단말기를 사용하거나 내부의 데이터를 접근할 수 없어야 한다.
- 플랫폼 인증 : 사용자와 서비스 제공자는 단말기가 불법 복제된 것이거나 도난당한 것은 아닌지, 사용하는데 있어 단말기의 상태가 안전한 상태인지 검증할 수 있어야 한다.
- 통신 기밀성 : 무선 통신로 상에 송수신 되는 데이터에 대해서 제 3자가 내용을 알 수 없도록 해야 한다.
- 데이터 기밀성 : 모바일 단말기 안에 저장된 데이터에 대해서 제 3자가 내용을 알 수 없도록 해야 한다.
- 데이터 무결성 : 단말기 내부에 저장된 데이터를 이전함에 있어 변조되지 않아야 하며, 이전시 통신 상에서도 데이터의 위조 및 변조가 되어서는 안 된다.
- 플랫폼 무결성 : 플랫폼에서 동작하는 운영체제와 같은 시스템 소프트웨어가 악성 코드에 의해 변경된다면 그 취약점으로 인해 사용자가 의도하지 않은 오작동 및 개인정보 유출을 가져올 수 있다. 따라서 플랫폼의 소프트웨어 및 하드웨어의 구성 상태에 대해 신뢰할 수 있도록 플랫폼 무결성을 제공해야 한다.
- 효율성 : 단말 간에 적은 양의 통신과 효율적인 암

호화 방법의 사용으로 안전성을 유지하며 데이터를 이전할 수 있어야 한다.

- 표준적합성 : 비밀 데이터의 이전은 실제 MTM이 적용되기 위해 필요한 연구로써 TCG 표준에서 제시한 것과 크게 다르지 않고 실제 적용할 수 있어야 한다.

Ⅲ. 관련 연구

본 장에서는 플랫폼 변경에 따른 기존 연구로써 TCG 표준에서 제안하고 있는 migration과 maintenance에 대해서 살펴보고, TMA를 사용한 이전 방안에 대해서 분석하고자 한다.

3.1 TCG 표준에서의 이전 방안

TCG 표준에서 제시한 비밀 데이터 이전 방안은 migration과 maintenance 두 가지 개념으로 설명할 수 있다. 이 두 가지 개념은 모두 하나의 TPM에서 다른 TPM으로의 비밀 데이터 이전에 관한 것을 다루고 있지만 이전하는 방법과 대상 데이터에 있어 차이점이 존재한다.

3.1.1 Migration

Migration은 플랫폼 내부 데이터의 안전한 이동을 위한 메커니즘으로, 데이터 보호에 사용된 키를 이전한다. 키는 크게 두 가지 타입(migratable, non-migratable)으로 나뉘는데, 이 두 가지 타입 중 migratable 키만이 migration을 통해 이전이 가능하며 non-migratable 키는 이전 불가능 하다. TPM에서 EK, SRK, AIK 등은 대표적인 non-migratable 키이며, 사용자에 의해 생성되는 키는 두 가지 타입 모두 가질 수 있다.

(1) 기본 메커니즘

Migration의 기본 메커니즘은 [그림 2]와 같으며 다음의 단계로 구성된다.

① 목적 TPM은 이전할 키를 안전하게 보호하기 위해 이전용 개인키/공개키 쌍을 생성한다.

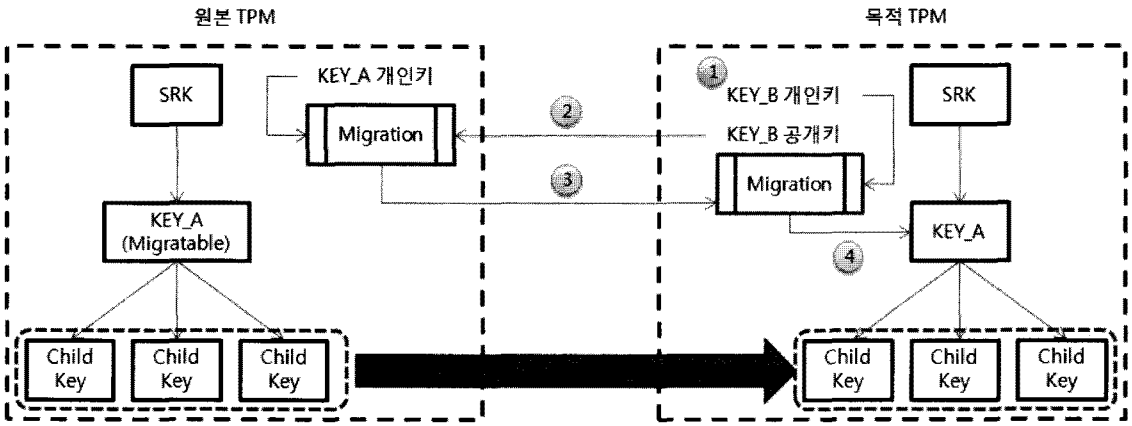
② 목적 TPM은 원본 TPM에게 생성한 이전용 공개키를 전달한다.

③ 원본 TPM은 이전할 개인키를 목적 TPM으로부터 전달받은 이전용 공개키로 다시 암호화하여 전송한다.

④ 목적 TPM은 이전용 개인키로 원본 TPM에게 받은 암호화된 데이터를 복호하고 그 키를 자신의 키 저장소에 저장한다.

(2) 분석

Migration은 AuthValue[1]를 사용해서 사용자를 인증하고, AIK를 사용한 전자서명으로 외부에 PCR 값을 reporting으로써 플랫폼을 인증한다. 그리고 플랫폼 내부의 데이터 이전을 위해 각각의 비밀 데이터에 대한



〔그림 2〕 Migration 기본 메커니즘

1) AuthValue는 TPM의 소유자가 자신이 TPM의 소유자임을 인증받기 위해 사용하는 비밀 값이다[18].

migration 키를 생성하여 데이터의 기밀성 및 통신 기밀성을 제공한다. 하지만 migration을 언급하고 있는 표준에서는 키 속성이 migratable로 설정된 키만 이전할 수 있도록 되어있기 때문에 non-migration의 속성을 가진 키는 이전이 불가능하다. 또한 사용자가 플랫폼을 교체하려면 안전한 저장소를 위해 사용되었던 키를 이전하기 위해 각각의 키에 대한 암호화가 따로 필요하기 때문에 효율성이 떨어진다. 따라서 TCG에서 언급하고 있는 migration 방법은 개개의 특정 키에 대한 이전은 용이할 수 있지만 키의 속성에 따라 불가능한 경우가 있으며 전체적인 이전 방안으로는 부적합하다.

3.1.2 Maintenance

Maintenance는 단지 키만 이전하는 것이 아니라 다른 TPM으로 기존의 모든 비밀 데이터를 이전하는 메커니즘으로 [그림 3]과 같다. Maintenance는 TPM의 필수적인 구현 사항은 아니며, 세부 사항 또한 플랫폼 제조사에게 위임하고 있다. Maintenance는 기존의 다른 비밀 데이터(예 : AuthValue, SRK)를 이전하기 위해 비휘발성 메모리에 접근해야 하며, migration이 불가능한 SRK등을 TPM 외부로 노출시켜야 한다. 플랫폼 제조사는 maintenance를 위해 사전에 각 제조 단말기 간에 동일한 maintenance 용 공개키를 가진다. 그리고 그 키를 사용하여 maintenance를 수행하게 된다. 때문에 maintenance는 동일한 플랫폼 제조사간에만 이루어 질 수 있고, SRK와 AIK, AuthValue 등도 이전이 가능하다.

(1) 기본 메커니즘

Maintenance가 이루어지는 과정은 [그림 3]과 같다.

① 플랫폼 제조사는 플랫폼 제조시 TPM 내부에 maintenance 용 공개키를 안전하게 저장한다.

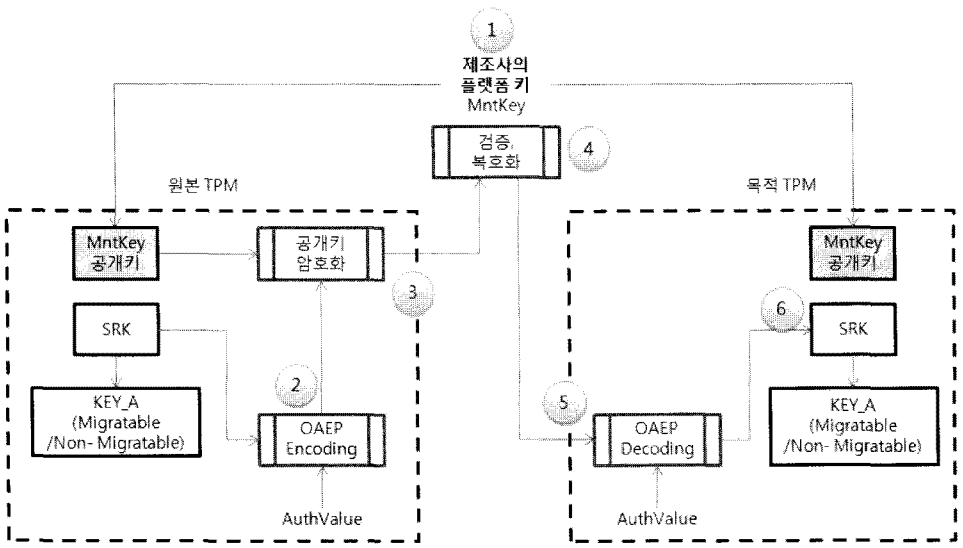
② Maintenance시에는 원본 플랫폼에서 이전할 데이터를 AuthValue에 기반을 둔 OAEP[2] 암호화를 사용해 스트림 암호화한다.

③ 그리고 다시 내부에 보관하고 있던 maintenance 용 공개키로 ②에서 암호화된 데이터를 2차적으로 다시 암호화하여 이전 데이터를 생성한다.

④ 만들어진 데이터를 제조사에게 전송하고 제조사는 해당 데이터의 유효성을 검증한다.

⑤ 검증 후 자신의 비밀키로 복호화하여 이전 목적 TPM에게 전달한다.

⑥ 목적 TPM은 AuthValue를 입력 받아 OAEP 복호화를 수행하여 데이터를 TPM에 로드한다.



[그림 3] Maintenance 기본 메커니즘

2) OAEP(Optimal Asymmetric Encryption Padding)는 주어진 입력값을 기반으로 특정 길이의 난수열을 생성하는 함수이다[8].

(2) 분석

Maintenance는 AuthValue를 사용해서 사용자를 인증하고, AIK를 사용하여 PCR 값을 reporting함으로써 플랫폼을 인증한다. Maintenance에서는 기밀성을 위해서 이중 암호화가 사용되는데, 이 중 암호화의 첫 번째로 데이터 기밀성을 위해 TPM의 인증정보인 AuthValue가 OAEP 암호화를 위해 사용되며 이는 실제 데이터에 대한 기밀성을 제공하게 된다. 두 번째로 이 암호화된 데이터를 TPM 제조시에 저장된 플랫폼 키로 다시 한 번 암호화를 수행하여 통신 기밀성을 가지는 이전 데이터를 생성한다. 따라서 중간의 제조사는 전송된 데이터에 대해 외부의 플랫폼 키에 의해 보호된 부분만 해독할 수 있으며 내부의 암호화된 데이터는 쉽게 알 수 없다. Maintenance는 데이터를 전송함에 있어 중간에 제조사를 거치게 되어 플랫폼과 제조사간에 두 번의 통신이 필요하고 사용자의 AuthValue로 OAEP 암호화된 데이터가 제조사에 노출될 수 있으며, 동일한 제조사간에만 수행될 수 있다는 단점을 지니게 된다. Migration과 달리 maintenance는 플랫폼 내부의 모든 암호키의 이전이 가능하지만 동일한 제조사간의 플랫폼으로써 공통된 maintenance용 플랫폼 키를 사용한다는 단점이 있으며, 사용자의 데이터가 암호화 되었지만 제조사를 통해 전달되어야 한다는 잠재적인 문제점을 내포하고 있다.

3.2 TMA를 통한 이전 방안

2005년 Ulrich Kuhn외 5명이 제안한 플랫폼 이전 방안[9]은 TMA(TPM Migration Authority)라는 신뢰기관을 이용하는데 TMA는 두 플랫폼의 EK 개인키/공개키 쌍 중 공개키에 대한 해쉬값을 서명하여 이전 인증서를 만든 후 이를 바탕으로 양 플랫폼이 서로를 식별하고 인증하는 방식을 이용한다. 일례로, 이전 대상 플랫폼은 자신의 EK 공개키를 이전할 데이터가 있는 플랫폼에 전송하게 되고 이전할 데이터가 있는 플랫폼에서는 인증서에 기록된 이전 대상 플랫폼의 EK 공개키 해쉬값과 비교 검증한 후 EK 공개키로 SRK를 암호화하여 데이터와 통신 기밀성을 유지하며 이전 데이터를 전송한다.

TPM 표준에서는 AIK를 생성하여 TPM에 대한 식별 키로 사용한다. EK를 TPM에 대한 식별자로 사용하지 않고 AIK를 사용하는 이유는 EK가 TPM에서 유일하기 때문에 EK 공개키 단독으로 사용할 때는 프라이버시 침해의 소지가 없지만, 다른 부가적인 개인 식별정보(일례로, AIK)와 결합하게 되면 EK로 개인을 식별할 수 있는 정보가 되기 때문에 프라이버시 침해의 우려가 있다[17].

Ⅳ. 제안방식

MTM에서의 키 관리 구조는 최상단에 SRK가 있어 하위 키들을 보호하고 있는 구조이다. 제안방식은 키 트리의 특성에 기인하여 상단의 SRK를 이전함으로써 플랫폼에서 사용하고 있는 암호화 키(Migratable/Non-Migratable)를 한 번에 이전하는 방안으로 TCG 표준의 maintenance 기능을 확장하여 플랫폼 내부의 SRK와 그 외의 non-migration의 속성을 가진 데이터를 이전한다. 본 제안방식은 사용자 인증 모듈인 USIM과 이전할 데이터가 있는 원본 MTM (MTM_S), 이전 대상 MTM인 목적 MTM(MTM_d), 외부인증기관인 CA로 구성된다. USIM은 사용자의 개인정보와 인증정보를 저장한 보안 모듈로써 단말기에 대한 사용자 인증을 수행할 때 사용되며 MTM은 USIM에 대해서 인증을 수행할 수 있다. 본 제안방식은 AuthValue만을 사용하여 사용자 인증 및 암호화를 수행하던 기존 방식에서 벗어나 사용자 인증 모듈인 USIM에 있는 사용자 개인정보와 비밀 값을 사용하는 Double Authentication[3]을 적용하여 AuthValue뿐 아니라 사용자 인증모듈인 USIM이 있어야 maintenance가 가능하도록 안전성을 강화시켰다.

4.1 시스템 계수

- * : 참여 객체(USIM, CA, MTM_S, MTM_d)
- ID_* : 개체 *의 식별자
- pw : USIM에 저장된 사용자의 패스워드
- pw' : 사용자가 입력한 패스워드
- $MigData$: SRK를 비롯한 이전할 비밀 데이터
- $ExData$: SRK에 의해서 암호화된 이전할 데이터
- r : 의사난수 (USIM내에서 사용자 개인정보를 바탕으로 생성)
- s : USIM의 의사난수 r과 USIM 안에 저장된 pw

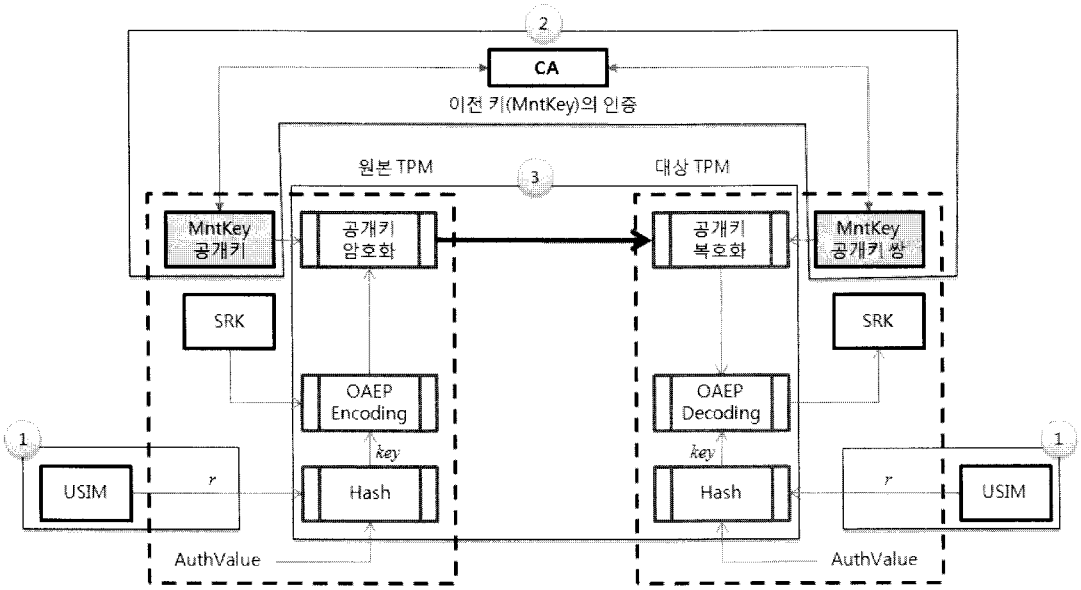
3) Double Authentication은 인증을 강화하기 위해 두 단계의 인증을 수행하는 방안으로 본 논문에서는 MTM의 AuthValue 인증과 사용자 인증모듈인 USIM에서의 사용자 비밀정보를 Double Authentication의 두 인증 요소로써 사용하였다.

를 XOR 연산하여 해쉬한 값

- s' : USIM의 의사난수 r과 사용자로부터 입력받은 pw`를 XOR 연산하여 해쉬한 값
- $Cert_*(key)$: *의 key에 대한 인증서
- $Sign_*[data]$: data에 대한 *의 서명 값
- $E_*[data]$: *의 공개키로 암호화한 데이터
- PR_{Mntkey}/PU_{Mntkey} : MTM_d에서 생성한 maintenance용 개인키/공개키
- PR_{AIK^*} : 플랫폼 *(MTM_S, MTM_d)을 대표하는 AIK 개인키/공개키 쌍 중 개인키
- $h(data)$: data의 일방향 해쉬 함수 값
- *AuthReq* : USIM 사용을 위한 사용자 인증 요청 메시지
- *pwReq* : 사용자에게 패스워드 입력을 요청하는 메시지
- *MntReq* : 이전용 키인 maintenance 키의 요청 메시지
- *MntKey* : maintenance를 위한 개인키/공개키 쌍
- *result* : USIM에 의한 사용자 인증 결과 (Authentication-Accept/Authentication-Fail)
- *TS* : 재전송 공격 방지를 위해 동기화된 MTM의 타임스탬프 값
- *KillSourceData* : 원본 MTM에서 사용자 데이터를 삭제

4.2 프로토콜

본 제안방식은 크게 사용자 인증 단계(①), 이전 키 설립 단계(②), Maintenance 완료 단계(③)로 나뉜다. 사용자 인증 단계는 maintenance에 앞서 원본 플랫폼의 소유자 인증을 하는 단계이며, 이전 키 설립 단계는 원본 플랫폼과 CA, 목적 MTM이 MntKey 개인키/공개키 쌍을 설립하는 단계이며 maintenance 완료 단계는 이전된 데이터에 대해서 검증하고 모든 프로토콜을 마무리하는 단계이다. 제안 방식의 전체적인 흐름은 [그림 4]와 같다. 본 프로토콜에서 MTM과 MTM간, MTM과 AS간의 통신은 메시지 무결성, 발신처 인증, 수신 부인방지를 위해 모든 메시지의 해쉬에 대한 전자서명을 인증기관으로부터 인증서를 받은 AIK 공개키로 제공하며 이 메시지는 재전송 공격 방지를 위한 동기화된 타임스탬프 *TS*를 포함하고 있다. *TS*는 구성 객체간의 동기화된 타임스탬프로써 전송 메시지에 대한 재전송 공격을 방지한다. 모든 메시지는 메시지를 전송 할 때마다 다른 *TS*를 사용하여야 하며 이는 전자서명에 의해 위조 및 변조로부터 보호되어 재전송 공격을 방지한다. 각 프로토콜 단계에서 명시적으로 공개키 검증 과정을 기술하지 않았지만 각 개체는 전자서명을 확인하기 위해 공개키에 대한 인증서를 검증하고, 전자서명을 확인하며 만일 올바르지 않은 공개키나 전자서명이 확인될 경우 프



[그림 4] 제안방식의 전체적인 흐름

로토콜을 중단한다고 가정한다.

4.2.1 사용자 인증 단계

사용자 인증단계는 사용자가 원본 MTM에 자신의 소유권을 인증하는 단계로 [그림 5]와 같다. 이 인증 단계에서 사용자는 기본 인증 정보인 AuthValue와 USIM을 사용하여 각각 사용자 인증을 수행한다. AuthValue를 사용한 MTM으로부터의 인증은 MTM을 사용하기 위해 필수적으로 이루어지는 것으로 사전에 이루어졌다고 가정한다.

- **Step 1** : 사용자는 먼저 AuthValue를 사용하여 MTM에게 인증받는다. 그리고 자신의 USIM을 원본 플랫폼에 삽입하여 USIM은 자신의 아이디와 함께 인증 요청 메시지와 난수 r과 TS, 그리고 전송 메시지에 대한 자신의 전자 서명을 MTM에게 전송한다.

$$ID_{USIM}, AuthReq, r, TS, \\ Sign_{USIM}[h(ID_{USIM}, AuthReq, r, TS)] \quad (1)$$

- **Step 2** : 원본 플랫폼의 MTM은 단말기의 입출력 장치를 통해 패스워드 입력을 요구하게 되고, 사용자로부터 입력된 패스워드 pw'를 받는다.

- **Step 3** : 원본 MTM은 사용자의 USIM으로부터 제공받은 난수 r과 사용자가 입력한 패스워드 pw'를 XOR 연산하고 해쉬하여 s'을 계산하고, 계산된 s'과 TS를 함께 MTM의 서명키로 서명한다. 그리고 USIM에게 자신의 아이디 함께 전송 메시지에 대한 서명 값을 전달하여 인증을 요구한다.

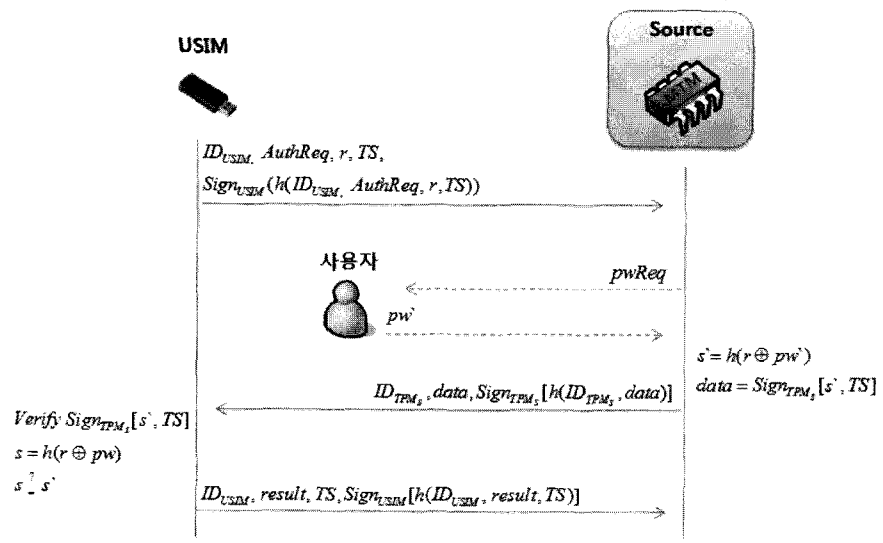
$$s' = h(r \oplus pw') \\ data = Sign_{TPM_s}[s', TS] \\ ID_{TPM_s}, data, Sign_{TPM_s}[h(ID_{TPM_s}, data)] \quad (2)$$

- **Step 4** : USIM은 전에 미리 생성한 난수 r과 내부에 저장된 사용자 인증정보인 패스워드 pw를 XOR 연산 하고 해쉬하여 s를 생성하고, 원본 MTM이 보낸 데이터 중 $Sign_{TPM_s}[s', TS]$에 있는 값 s'를 추출하여 USIM이 계산한 s와 일치하는지를 검증한다. 이 결과로 정당한 사용자에 의한 패스워드 입력을 검증하여 사용자 인증 여부를 판정하고 그 결과 result 값을 원본 MTM에게 전달한다.

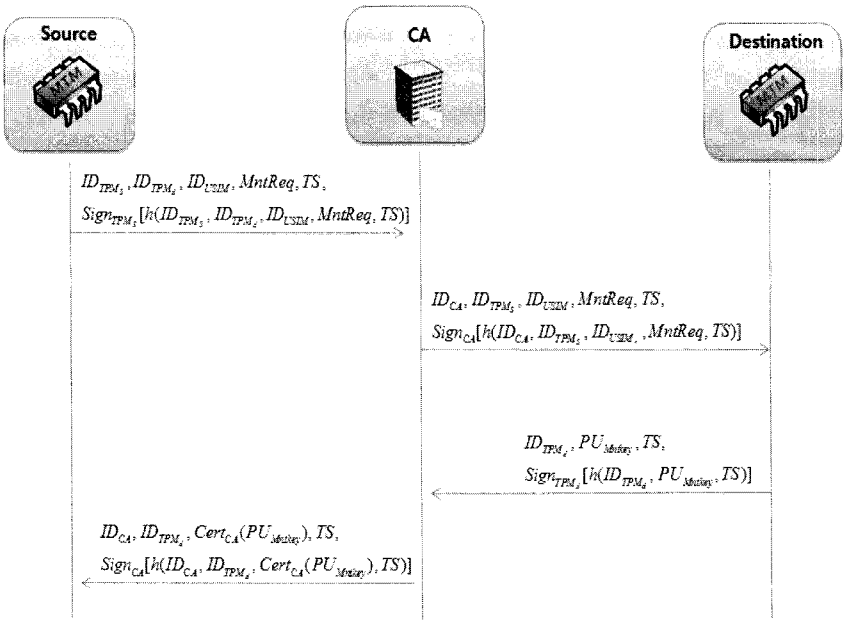
$$s = h(r \oplus pw) \\ s \stackrel{?}{=} s' \quad (3)$$

4.2.2 이전 키 설립 단계

이전 키 설립 단계는 [그림 6]과 같이 인증 기관 CA를 통해서 maintenance 용 키를 설정하는 단계이다. MTM_d는 이미 MTM_s의 아이디를 알고 있다고 가정한



[그림 5] 사용자 인증 단계



〔그림 6〕 이전 키 설립 단계

다. MTM과 CA간의 통신에서 maintenance 키를 설정함에 있어 무결성 및 발신자 인증을 제공하기 위해 전자서명을 사용한다. 전자서명 확인에 사용되는 각 플랫폼의 AIK 공개키는 CA에 의해 인증서가 발급된 키로써 상호간에 공개키에 대한 검증을 수행하여 메시지의 유효성을 판정한다고 가정한다.

- **Step 1** : 원본 MTM은 자신의 아이디와 목적 MTM의 아이디, USIM의 아이디, 그리고 maintenance 키 요청 메시지 *MntReq*와 *TS*와 자신의 서명키로 메시지에 대한 서명을 생성하여 CA에게 전송한다.

$$ID_{TPM_S}, ID_{TPM_d}, ID_{USIM}, MntReq, TS, \quad (4)$$
$$Sign_{TPM_S}[h(ID_{TPM_S}, ID_{TPM_d}, ID_{USIM}, MntReq, TS)]$$

- **Step 2** : 인증기관인 CA는 원본 플랫폼의 *MntReq*에 대한 MTM_S의 서명을 검증하여 확인하고, 수신된 *MntReq*에 대해서 인증기관의 서명을 추가하여 MTM_d에게 정당한 요청 메시지임을 전송한다.

$$ID_{CA}, ID_{TPM_S}, ID_{USIM}, MntReq, TS, \quad (5)$$
$$Sign_{CA}[h(ID_{CA}, ID_{TPM_S}, ID_{USIM}, MntReq, TS)]$$

- **Step 3** : 목적 MTM은 MntKey 개인키/공개키 쌍 $(PR_{Mntkey}, PU_{Mntkey})$을 MTM이 자신의 내부에서 안전하게 생성하고, 자신의 아이디와 이전용 공개키(PU_{Mntkey}), 그리고 전체 메시지에 대한 서명 값을 함께 CA에게 전달한다.

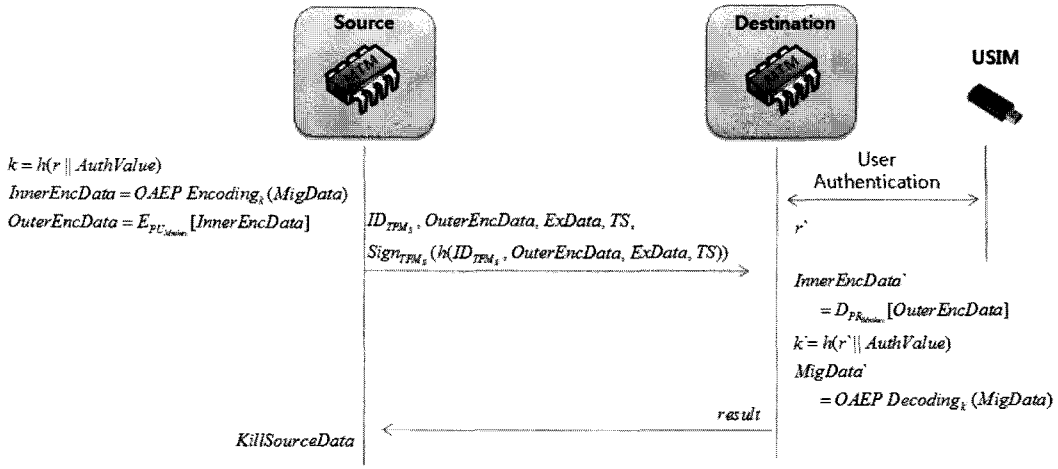
$$ID_{TPM_d}, PU_{MntKey}, TS, \quad (6)$$
$$Sign_{TPM_d}[h(ID_{TPM_d}, PU_{MntKey}, TS)]$$

- **Step 4** : CA는 목적 MTM에게서 온 데이터의 서명을 검증하고, 이전용 공개키 PU_{Mntkey}에 대한 인증서 $Cert_{CA}(PU_{MntKey})$를 발급하여 원본 MTM에게 전송한다.

$$ID_{CA}, ID_{TPM_D}, Cert_{CA}(PU_{MntKey}), TS, \quad (7)$$
$$Sign_{CA}[h(ID_{CA}, ID_{TPM_D}, Cert_{CA}(PU_{MntKey}), TS)]$$

4.2.3 Maintenance 완료 단계

Maintenance 완료 단계는 [그림 7]과 같이 목적 MTM에서 사용자 인증 후에 전송된 데이터를 복호하여 자신의 데이터로 만드는 과정이다. 이전하는 데이터는 SRK가 포함되어있는 *MigData*와 *ExData*이다. 이 후 MTM_S에 maintenance가 완료된 사실을 알려야 하며 MTM_S는 그 사실을 확인한 후에 자신의 내부에 있는 데이터를 삭제해야 한다.



〔그림 7〕 Maintenance 완료 단계

• **Step 1** : 사용자는 AuthValue로 MTM_d으로부터 인증받고 USIM을 사용하여 사용자 인증을 수행한다[4]. 이 과정은 MTM_S에서 수행하는 4.2.1의 사용자 인증 단계와 같은 과정이다. 이 때 USIM이 MTM_S에 제공하는 난수 r은 4.2.1에서의 r값과 같은 값이다.

• **Step 2** : MTM_S은 사용자 인증 과정에서 USIM으로부터 전송 받은 r과 AuthValue를 함께 해쉬하여 OAEP 암호화에 사용될 입력 키 값 k를 생성한다. 그리고 1차적으로 MigData를 암호화하고($InnerEncData$), 2차적으로 PU_{Mntkey}로 다시 암호화하여 최종 데이터($OuterEncData$)를 생성한다.

$$k = h(r, AuthValue) \quad (8)$$
$$InnerEncData = OAEP\ Encoding_k(MigData)$$
$$OuterEncData = E_{PU_{Mntkey}}[InnerEncData]$$

이후, 자신의 아이디와 $ExData$, 그리고 서명을 함께 MTM_d에게 전송한다.

$$ID_{TPM}, OutterEncData, ExData, TS, \quad (9)$$
$$Sign_{TPM_S}(h(ID_{TPM}, OutterEncData, ExData, TS))$$

• **Step 3** : MTM_d은 자신이 생성한 maintenance 용 비밀키를 사용하여 원본 플랫폼으로부터 받은 이전 데이터를 복호한다. 그리고 USIM을 통해 사용자 인증에 따라 입력받은 r과 AuthValue를 이용하여 만든 입력 키 값 k로 OAEP 복호화을 수행하여 최종적으로 MigData를 얻는다.

$$k = h(r', AuthValue) \quad (10)$$
$$InnerEncData = D_{PR_{Mntkey}}[OuterEncData]$$
$$MigData = OAEP\ Decoding_k(MigData)$$

• **Step 4** : 목적 MTM은 maintenance 결과를 원본 플랫폼에 전송한다. 원본 MTM은 목적 MTM으로부터 maintenance가 완료된 것을 확인한 후 자신이 가지고 있던 SRK를 비롯한 비밀 데이터를 삭제한다($KillSourceData$).

V. 제안방식 분석

본 제안방식은 비밀 데이터를 이전하는 두 개념인 migration과 maintenance에서 SRK 이전이 가능한 maintenance 방안을 확장하여 제안하였다. Maintenance는 단지 한 두 개의 키를 이전하는 것이 아니라 플랫폼에 있는 비밀 데이터를 모두 이전하는 개념이다. 본 제안방식에서는 maintenance를 위해 상호 간에 이전 키 설립이 가능하도록 하였고, 그것을 신뢰기관인 CA가 중재한다.

Maintenance를 위해서는 사용자가 자신의 인증 모듈인 USIM을 사용해서 사용자 인증과 TPM에 대한 소유자 인증을 함께 받아야 maintenance가 가능하다.

또한 TCG 표준에 명시된 제조사가 maintenance 용 공개키를 미리 설정하고 maintenance 개인키를 소유함으로써 반드시 제조사를 거치게 되는 번거로움을 없애

4) MTM_d에서의 AuthValue는 MTM_S에서의 AuthValue와는 다른 값으로 각각의 MTM으로부터 인증 받기위한 독립적인 비밀 값이다.

〔표 1〕 기존방식과 제안방식의 비교 〔X : 제공 못함, △ : 보통, ○ : 안전, ◎ : 매우안전〕

<table>
<tr><th colspan="2" rowspan="2">구분</th><th colspan="2">표준</th><th rowspan="2">TMA를 사용한
Migration 방식</th><th rowspan="2">제안방식</th></tr>
<tr><th>Migration</th><th>Maintenance</th></tr>
<tr><td rowspan="2">인증</td><td>사용자 인증</td><td colspan="3">○
(AuthValue)</td><td>◎
(AuthValue, pw)</td></tr>
<tr><td>플랫폼 인증</td><td colspan="4">◎
(AIK)</td></tr>
<tr><td rowspan="2">기밀성</td><td>통신 기밀성</td><td rowspan="2">○
(Migration 키)</td><td>○
(플랫폼 키)</td><td rowspan="2">○
($EK_{Destination}$)</td><td>○
(PU_{MntKey})</td></tr>
<tr><td>데이터 기밀성</td><td>○
(AuthValue)</td><td>◎
(AuthValue, s)</td></tr>
<tr><td rowspan="2">무결성</td><td>데이터 무결성</td><td colspan="4">◎
(전자서명)</td></tr>
<tr><td>플랫폼 무결성</td><td colspan="4">◎
(PCR Reporting)</td></tr>
<tr><td colspan="2">효율성</td><td>X
(모든 키에 대한
암호화)</td><td>△
(동일 제조사간의
SRK 이전)</td><td>○
(EK를 사용한
SRK 이전)</td><td>○
(다른 제조사 간의
SRK 이전)</td></tr>
<tr><td colspan="2">표준 적합성</td><td colspan="2">◎
(TCG 제안)</td><td>X
(EK를 사용한
SRK의 이전)</td><td>△
(다른 플랫폼 제조사간의
Maintenance)</td></tr>
</table>

기 위해 본 제안 방식에서는 maintenance 키를 인증해 주는 신뢰기관 CA를 두고 maintenance 키를 양 플랫폼 사이에서 설립하도록 하여 다른 플랫폼 사이에서도 신뢰기관인 CA를 통해 maintenance를 할 수 있도록 하였으며, 사용자는 USIM을 통해 보다 안전한 maintenance 기능을 이용할 수 있게 되었다. 따라서 본 제안 방식은 기존의 migration의 장점인 다양한 제조사 간의 플랫폼 간에도 안전한 비밀 데이터의 이동이 되는 점과 일반적으로 이전할 수 없는 non-migratable한 키를 이동 할 수 있는 점을 조합하고, USIM과 MTM을 사용하여 이동성 있는 신뢰 모듈과 고정적인 신뢰모듈을 상호 연동함으로써 안전하게 데이터를 이전 할 수 있는 장점이 있다. 하지만 CA를 이용하여 메시지를 중재하는 구조는 양 객체간의 데이터 이동을 안전하게 수행할 수 있도록 하는 반면 전체적인 프로토콜이 무거워 질 수 있는 단점이 될 수 있다. 제안 방식을 앞서 언급한 보안 요구사항에 따라 분석하면 다음과 같다.

- 인증 : 모바일 단말기는 이전하고자 하는 상대 단말기에 대해서 인증을 해야 한다. 본 제안방식에서는 두 가지의 인증이 존재한다. 첫 번째는 사용자에 대한 인증이며, 두 번째는 모바일 단말기에 대한 인증이다. 본 제안방식에서는 사용자 인증을 위해 기존의 AuthValue외에 사용자 인증 모듈인 USIM을 사용하였다. USIM에는 사용자의 인증정보인 pw가 안전하게 저장되어 있으며, USIM을 통해서 사용자를 인증하고 TPM을 사용하여 maintenance를 위한 데이터를 생성하기위해 USIM과 AuthValue 모두를 사용한다. 또한 플랫폼 인증은 해당 플랫폼이 정당한 플랫폼인지 인증하는 것으로 외부 인증기관에 의해 인증된 AIK를 사용함으로써 제공되며, AIK에 의한 전자서명은 공개키 검증으로 발신처 인증 및 메시지 인증, 전송 부인방지를 제공한다.
- 기밀성 : 플랫폼간 이전에 의해 데이터가 MTM 외부로 노출될 수 있으므로 이전 대상인 데이터와 그 데이터를 보호하고 있던 키들에 대해서도 기밀성을 유지해야 한다. 본 제안방식에서는 안전한 저장소를 이루는 핵심인 MTM 내부의 SRK를 이전하기 위해 USIM과 AuthValue를 사용한 Double Authentication을 제안하였다. 이 두 요소는 키 생성을 위한 해쉬값의 입력으로 사용되며 생성된 키는 OAEP 암호화 키 로 사용되어 전송되는 데이터의 기밀성을 제공한다. 또한 maintenance 키인 PU_{MntKey}는 목적 플랫폼에 데이터를 안전하게 전달하기 위해 통신 기밀성을 제공하는데 사용된다.
- 무결성 : MTM이 적용된 플랫폼은 PCR에 의해 플랫폼 무결성이 측정되며, 데이터에 대한 무결성은 전자서명으로 제공된다. 플랫폼 무결성은 기본적으로 PCR을 통해 제공된다. PCR에 대한 검증 값을 MTM이 외부 인증기관으로부터 인증 받은

AIK 전자 서명키를 사용하여 PCR에 대한 서명으로 제공하면 제 3자는 플랫폼의 PCR 값을 검증하여 플랫폼에 대한 무결성을 PCR에 대한 MTM의 AIK 전자서명으로 제공된다. 또한 AIK에 의한 서명은 통신 데이터의 무결성을 제공한다.

- 효율성 : 본 제안방식은 MTM내의 안전한 저장소를 제공하는 SRK를 이전하는 maintenance 개념을 사용하여 각각의 비밀 데이터에 대한 migration보다 효율적으로 비밀 데이터를 이전하도록 하였다. SRK를 이전하는 것은 안전한 저장소에 접근할 수 있는 전체적인 권한을 얻는 것으로 안전한 저장소의 암호화된 데이터에 모두 접근할 수 있다. 또한 기존의 maintenance가 플랫폼 키를 미리 삽입하여 동일한 제조사간에만 가능하도록 하였지만 본 방식에서는 제 3의 신뢰기관인 CA를 두어 maintenance 키를 목적 MTM이 생성해서 원본 MTM에 전달하도록 하여 다른 제조사간에도 maintenance를 할 수 있도록 하였다.
- 표준 적합성 : 본 제안방식은 TCG 표준의 maintenance에 기반한 방식으로 non-migratable 속성을 가진 SRK의 이전을 통해 비밀 데이터를 이전한다. Maintenance는 TCG가 구체적인 구현사항을 명시적으로 표준화하지 않고 단지 기본적인 틀만 제공하고 있다. 본 논문에서의 제안은 표준에서의 maintenance가 동일한 플랫폼에만 가능하다는 것을 확장하기 위해 미리 삽입되는 플랫폼 키 대신 이전을 위한 maintenance 키를 플랫폼 상호간에 설립할 수 있도록 하여 TMA를 이용한 기존 방식에서의 EK를 사용하여 SRK를 이전 시키는 방안에 비해 표준 적합성을 가진다.

Ⅵ. 결 론

신뢰 컴퓨팅은 향후 유비쿼터스 환경에 맞추어 중요한 보안기술로서 성장할 가능성이 있다. 이에 본 논문에서는 현재까지 개발된 신뢰 컴퓨팅과 관련된 기술 중에 모바일 환경의 MTM에 초점을 두었으며 구현과 적용의 문제점에 따른 방안을 제시하였다. 언급된 적용과 구현에 관한 문제는 MTM이 우수한 보안성을 제공하지만 그러한 만큼 다른 칩 제조사간의 상호 호환성 문제, 플랫폼 이전 및 비밀 데이터 이전에 있어 유연하지 못한 점 등이다[1,9]. 따라서 이러한 문제점을 해결하고자 안전한 저장소의 핵심이 되는 SRK를 이전할 수 있는 새로운 maintenance 방법을 본 논문에서 제안하였다. 본 제안은 사용자 인증을 위해 MTM에 대한 인증과 함께 USIM을 사용함으로써 Double Authentication을 적용하였고, TCG 표준에 기술된 maintenance 단점에 대한 해결책으로 maintenance 키를 인증해주는 별도의 신뢰 기관을 둠으로써 양 플랫폼 간의 실질적인 데이터 교환이 이루어지도록 하였다. 향후 연구 방향으로는 maintenance 뿐만 아니라 MTM의 상호 호환성과 데이터 이전 및 플랫폼의 MTM에 의존적인 sealed data[18]의 이전을 위해 MTM과 사용자 인증을 제공하는 USIM과의 연동을 통한 보안 기술에 관한 연구가 이루어 져야 할 것이며 이론적인 연구와 함께 실제 적용을 위한 구현을 병행하여 실질적인 평가가 이루어 져야 할 것이다.

참고문헌

[1] A. Sadeghi, Marcel Selhorst, Christian Stuble, Christian Wachsmann, Marcel Winandy, "TCG Inside? - A Note on TPM Specification Compliance," *STC'06*, 2006.

[2] A. K. Ghosh, Tara M. Swaminatha, "Software security and privacy risks in mobile e-commerce," *Communications of the ACM*, Vol 44, No 2, pp.51-57, 2001.

[3] B. Halpert, "Mobile Device Security," *InfoSecCD Conference'04*, 2004.

[4] C. Covey, Mark Redman, Thomas Tkacik, "An Advanced Trusted Platform for mobile phone devices," *Information Security Technical Report*, Vol 10, pp.96-104, 2005.

[5] E. Gallery, C. J. Mitchell, "Trusted Mobile Platforms," *Foundations of Security Analysis and Design IV*, LNCS 4677, pp.282-323, 2007.

[6] J. Lindqvist, Laura Takkinen, "Privacy Management for Secure Mobility," *WPES'06*, 2006.

[7] J. Reid, Juan M. Gonz´alez Nieto, Ed Dawson, Eiji Okamoto, "Privacy and Trusted Computing," *DEXA'03*, 2003.

[8] M. Bellare and P. Rogaway, "Optimal asymmetric encryption," *Advances in Cryptology -*

Eurocrypt'94, Springer-Verlag, 92-111, 1994.

[9] Ulrich Kuhn, Klaus Kursawe, Stefan Lucks, "Secure Data Management in Trusted Computing," *CHES 2005*, LNCS 3659, pp.324-338, 2005.

[10] 김무섭, 신진아, 박영수, 전성익, "모바일 플랫폼용 공통보안핵심 모듈 기술," *정보보호학회지*, 제 17권, 제 3호, pp.7-17, 2006.

[11] 김영수, 박영수, 박지만, 김무섭, 김영세, 주홍일, 김명은, 김학두, 최수길, 전성익, "신뢰 컴퓨팅과 TCG 동향," *전자통신동향분석*, 제 22권 제 1호, pp.83-96, 2007.

[12] Trusted Computing Group, "Backgrounder," 2006.

[13] Trusted Computing Group, "Mobile Phone Work Group Use Cases," 2005.

[14] Trusted Computing Group, "Mobile Trusted Module Specification FAQ," 2006.

[15] Trusted Computing Group, "Mobile Trusted Module Specification General Overview FAQ," 2007.

[16] Trusted Computing Group, "Mobile Trusted Module Specification Technical Overview FAQ," 2007.

[17] Trusted Computing Group, "TCG Specification Architecture Overview," Revision 1.4, 2007.

[18] Trusted Computing Group, "TCG TPM Specification Version 1.2 Revision 103," 2007.

〈著者紹介〉

강 동 완 (Dong-Wan Kang) 학생회원
2007년 2월 : 순천향대학교 정보기술공학부 졸업
2007년 3월~현재 : 순천향대학교 컴퓨터학과 석사과정
<관심분야> 정보보호, 신뢰컴퓨팅, TPM, 키 관리

이 임 영 (Im-Yeoung Lee) 종신회원
1981년 2월 : 홍익대학교 전자공학과 졸업
1986년 2월 : 오사카대학 통신공학전공 석사
1989년 2월 : 오사카대학 통신공학전공 박사
1985년~1994년 : 한국전자통신연구원 선임연구원
1994년~현재 : 순천향대학교 컴퓨터학부 교수
<관심분야> 암호이론, 정보이론, 컴퓨터 보안

한 진 희 (Jin-Hee Han) 정회원
1997년 2월 : 숭실대학교 정보통신공학과 졸업
1999년 2월 : 광주과학기술원 정보통신공학과 석사
1999년 6월~현재 : 한국전자통신연구원 무선보안응용연구팀 선임연구원
<관심분야> 정보보호, 스마트 카드, USIM, 무선 보안 기술, TSS(TCG Software Stack)

전 성 익 (Sung-Ik Jun) 정회원
1985년 2월 : 중앙대학교 전자계산학과 졸업
1987년 2월 : 중앙대학교 전자계산학과 석사
1987년~현재 : 한국전자통신연구원 책임연구원
2003년~현재 : 한국전자통신연구원 무선보안응용연구팀 팀장
<관심분야> 정보보호, 스마트 카드, USIM, 무선 보안, TPM