모바일 지갑 세션보호를 위한 디바이스 페어링 기술 성능평가

마건일[O1], 이형찬[1], 기현식[2], 최대선[3], 진승헌[3], 이정현[1]

[1]숭실대학교 컴퓨터학부, [2]삼성전자, [3]한국전자통신연구원

gima@ssu.ac.kr, lee.hyeongchan@ssu.ac.kr, there99@naver.com, sunchoi@etri.re.kr, jinsh@etri.re.kr, jhyi@ssu.ac.kr

Performance Evaluation of Device Pairing Techniques for Establishing Secure Session Using Mobile Wallet

Gun Il Ma[O1], Hyeong Chan Lee[1], Hyunsik Ki[2], Daeseon Choi[3], Seung Hun Jin[3], Jeong Hyun Yi[1]

[1]Soongsil University, School of Computing, [2]Samsung Electronics

[3]Electronics and Telecommunications Research Institute

요 약

높은 이동성 및 휴대성을 갖는 모바일 디바이스의 기술적 발전은 사용자로 하여금 보다 높은 수준의 통합된 편의 기능 제공이 요구되고 있다. 이러한 예로 기존 물리적 지갑에 보관하던 플라스틱 신용카드, 멤버쉽 카드, 신분증 등의 개인정보를 모바일 디바이스 안에 저장 관리하는 모바일 지갑 서비스가 현실화 되고 있다. 모바일 지갑을 통한 상거래 서비스를 이용할 시 디바이스에 저장된 각종 개인정보가 근거리 무선통신 기술을 통해 다른 모바일 기기나 지불서버에 전달되는 데, 이 무선전송 구간은 근원적으로 많은 보안 취약점을 갖고 있다. 따라서 본 논문에서는 모바일 지갑 응용 서비스에 모바일 기기간 안전한 키 설정 시 공개키 인증서를 활용하지 않고 두 기기간 공유키 검증을 할 수 있는 세션 키 검증 기술들을 분석하고, 해당기술들을 구현하여 모바일 지갑 결제 테스트베드에 포팅한 실험 결과를 통한 성능분석 결과를 제시한다. 본 성능평가를 통해 향후 다양한 모바일 기기 특성에 따른 최적의 세션 키 공유 키 검증 방법 선택 시 유용한 근거자료로 활용할 수 있을 것으로 기대된다.

1. 서 론

스마트 폰 보급과 사용이 빠른 속도로 확대되면서 모바일 디바이스를 활용한 다양한 서비스가 소개되고 있다. 그 가운데 모바일 지갑 서비스는 사용자의 신용카드, 포인트 카드, 멤버쉽 카드 등을 비롯한 모든 디지털 ID 정보를 스마트폰에 저장한 후, 이를 통해 다양한 지능형 지불 및 개인화 서비스가 가능하도록 해 주는 응용서비스이다. 이는 하나의 신용카드 정보만을 휴대폰에 탑재하여 휴대폰으로 결재를 제공해주는 모바일 뱅킹 서비스와 달리, 개인의 프로화일 정보를 활용하고, 최적의 지불수단을 추천해 주는 보다 진보된 지불서비스라고 할 수 있다.

이러한 모바일 지갑 서비스는 퍼베이스브 연결 (Pervasive Connectivity) 기술을 통해 다양한 근거리 범위의 무선 디바이스들과 연결을 통해 제공될 것이다. 하지만 무선통신은 근원적으로 많은 보안 취약점을 가지고 있다. 따라서 모바일 지갑 서비스가 실용화되기 위해서는 모바일 디바이스 간 안전한 세션 보호 기술이 선행적으로 제공되어야 한다.

본 논문에서는 모바일 디바이스 간 세션 키 교환 기술로 Diffie-Hellman 프로토콜을 사용하고, 세션 키의 검증 방법으로 사람이 직접 세션 키 공유 여부를 확인 할 수 있도록 기존 기술들[2][3][4]을 직접 재연실험을 통해 실현가능성 여부를 판단하기 위한 실험 결과를 보고한다.

또한 실험 결과의 분석 및 평가를 통해 모바일 지갑 서비스에 적합한 세션보호 기법에 대한 논의한다.

본 논문의 구성은 다음과 같다. 2장 관련연구에서는 모바일 지갑의 페어링에 사용될 프로토콜에 대해 설명한다. 3장에서는 모바일 지갑의 세션 키 검증기술에 대해 설명한다. 4장에서는 모바일 지갑 결제 시스템을 통해 세션 키 검증 기술 구현을 설명한다. 5장에서는 4장에서 보인 기술의 성능을 비교 분석한다. 6장에서는 본 논문에서 보인 세션 키 검증 기술의 성능평가를 토대로 키 검증 기술의 실현방향을 제시하고 결론을 맺는다.

2. 관련연구

본 장에서는 모바일 지갑 서비스에 활용되는 관련 기반 기술들에 대해 간략하게 살펴본다.

2.1 Diffie-Hellman 프로토콜

Diffie-Hellman (DH) 프로토콜 방법에서는 양쪽 통신주체가 키분배센터없이 대칭 세션 키를 생성한다. 대칭 키를 만들기 전에 양쪽은 두 개의 수 p와 g를 선택해야 한다. 여기서 p는 매우 큰 소수로서 1024 비트의 길이를 갖는다. 두 번째 수 g는 군<Z_p*, x>의 원소로서 위수가 $p-1$인 생성자이다. 이 두 가지 정보(군과 생성자)는 비밀로 간직할 필요가 없다. 이 두 값을 안전하지 않은 일반 통신채널을 통해서 전송한다. 다시 말해 공개되어도 무방하다. A와 B가 DH 프로토콜을 이용하여 키를 공유하는 과정은 (그림 1)과 같다.

$$A \rightarrow B: y_a = g^a \bmod p$$
$$A \leftarrow B: y_b = g^b \bmod p$$

(그림 1) DH 키 공유 방식

1. A는 자신의 비밀 키 a를 $0 \le a \le p$-1 안에서 선택하고 A의 공개키 y_a을 B에게 보낸다. B또한 자신의 비밀 키 b를 $0 \le b \le p$-1 안에서 선택 한 후 B의 공개키 y_b을 A에게 보낸다.
2. A와 B는 아래와 같이 대칭키 K를 계산한다.
$$A: K = (y_b)^a \bmod p = g^{ab} \bmod p$$
$$B: K = (y_a)^b \bmod p = g^{ab} \bmod p$$
3. B는 a값을 모르고, A는 b값을 모르면서 두 사람이 동일한 대칭키 K 값을 얻게 되었다.

DH 프로토콜은 중간자 공격 (Man-In-The-Middle)의 약점이 있다. 공격자 E는 이 프로토콜을 공격하기 위해서 a와 b값을 구할 필요가 없다. E는 두 개의 키를 만들어서 A와 B를 속일 수 있다. 하나의 키는 자신과 A사이에 사용할 것이고, 다른 하나는 자신과 B 사이에 사용할 키이다. 중간자 공격과정은 (그림 2)와 같다.

$$A \rightarrow E: y_a = g^a \bmod p$$
$$A \leftarrow E: y_e = g^e \bmod p$$
$$E \rightarrow B: y_e = g^e \bmod p$$
$$E \leftarrow B: y_b = g^b \bmod p$$

(그림 2) DH 중간자 공격

1. A는 비밀 키 a를 선택한 다음 A의 공개키 y_a를 B에게 보낸다. 이때 E는 중간에서 이 값을 가로챈다. 이후 E는 e를 선택한 다음 y_e을 A와 B에게 보낸다.
2. B는 비밀 키 b를 선택한 다음 B의 공개키 y_b를 A에게 보낸다. 이때 E는 중간에서 이 값을 가로챈다.
3. A와 E는 $g^{ae} \bmod p$를 계산하여 공유키로 갖는다. 또한 E와 B는 $g^{be} \bmod p$를 계산하여 공유키로 갖는다. 하지만 A와 B는 서로 다른 공유키를 갖게 되는 사실을 알지 못한다.

2.2 Station-to-Station 프로토콜

Station-to-Station (STS) 프로토콜은 DH 에 기반을 둔 방법이다. 이 프로토콜은 A와 B사이의 세션 키를 만들기 위해 공개 키 인증서를 이용한 디지털 서명을 사용한다. (그림 3)에 이러한 절차를 나타내고 있다.

$$A \rightarrow B: y_a = g^a \bmod p$$
$$B: K = (y_a)^b \bmod p$$
$$A \leftarrow B: y_b = g^b \bmod p, E_k(S_B(A|y_a|y_b)), B's\ Certificate$$
$$A: K = (y_b)^a \bmod p$$
$$A \rightarrow B: E_k(S_A(B|y_a|y_b)), A's\ Certificate$$

(그림 3) Station-to-Station 키 합의 방법

1. A는 공개키 y_a를 B에게 보낸다.
2. B는 세션 키 K를 계산하고 A의 ID, y_a, y_b를 이어 붙인 후 비밀 키 b로 서명을 한다. 서명은 세션 키로 암호화한다. B는 y_b, 서명, 자신의 공개 키 인증서를 A에게 보낸다.
3. A는 세션 키 K를 계산한 뒤에, B의 서명이 검증되면 B의 ID, y_a, y_b를 이어 붙인 후 자신의 비밀 키 a로 서명한다. 이 서명은 세션 키로 암호화 한다. A는 서명, 자신의 공개 키 인증서를 B에게 보낸다.
4. B는 A의 서명이 검증되면 세션 키를 받아들인다.

STS 프로토콜은 중간자 공격을 막는다. E는 y_a을 가로챈 뒤에 자신의 y_b를 A에게 보내어 그것이 B에게서 온 것처럼 꾸밀 수 없다. E는 B의 비밀 키를 알 수 없기 때

문에 서명을 생성할 수 없다. 만약 서명을 꾸몄다고 하더라도 인증서에 들어있는 B의 공개키로 검증이 되지 않는다. 마찬가지로 E는 A의 비밀 키 또한 모르기 때문에 A가 보낸 메시지에 대한 서명도 위조할 수 없다.

하지만 이 기법은 PKI (Public Key Infrastructure)를 요구하게 되므로, TTP (Trusted Third Party)없이 키 공유를 하고자 했던 DH 알고리즘의 기본 사상에도 위배될 뿐만 아니라, 이동성이 잦고, 송수신 단말을 동일 사용자가 인증 가능한 모바일 디바이스 간 세션보호 기술로는 적합하지 않다.

3. 근거리 세션 키 검증 기술

모바일 지갑의 근거리 무선통신에서 안전한 통신을 위해 DH 프로토콜을 사용하여 세션 키 교환을 수행할 때 잘 알려진 취약점은 중간자 공격이다. 중간자 공격을 막기 위해서는 세션 키 교환 후 키 검증과정을 거치는 페어링 기술을 사용하여야 한다.

본 논문에서는 송수신자가 사이버 공간에서 서로 떨어져 있어 PKI와 같은 TTP를 활용해야 하는 기존 방법과 달리, 모바일 지갑 응용에서는 송수신 단말을 사용자가 직접 물리적으로 확인 가능하므로 보다 간단하고 편리한 세션 키 인증 방법이 가능해 진다. 이에 해당하는 기술로 Seeing is Believing (SiB) [2], Hearing is Believing (HiB) [3], Blinking is Believing (BiB) [4]가 있는데, 본 장에서는 각 기술들에 대해 간략하게 살펴본다.

3.1 SiB 기법

사용자가 직접 확인하여 승인을 하는 인증 방법을 이용하여 PKI 없이 중간자 공격을 방어하려면, 계산된 공유키를 눈으로 볼 수 있는 방법으로 표현하는 기술이 필요하다. SiB 기술은 계산된 공유키를 눈으로 볼 수 있는 방법으로 표현하기 위해 공유키의 해쉬된 데이터를 바코드 형태로 디스플레이에 표현하는 방법이다. 본 논문에서는 바코드 형태의 표현방법을 좀 더 개선하여 4가지의 다른 색상을 가지는 그리드 형태의 표현방법을 사용하였다. 8x8 크기의 그리드는 공유된 세션 키 데이터의 해쉬 값 일부를 미리 지정된 색상 값으로의 매핑을 통해 만들어진다.

3.2 HiB 기법

HiB은 SiB와 유사하지만 카메라로 바코드를 인식해서 동일한 공유키를 소유여부를 판단하는 과정에서 바코드 인식률이 떨어지는 단점을 보완하여 고안된 방법이다.

HiB 방법은 계산된 공유키를 일정하지 않은 단어나 문자열로 변환한 후 변환된 문자열을 스피커를 통해 표현되면 마이크로 그 소리를 받아들여 각 디바이스의 공유키를 확인 하거나 두 디바이스에서 같은 문장의 소리를 들리게 한 후 사람이 직접 들어 생성된 두 문장이 같은지를 확인하여 인증하는 방식이다. 본 논문에서는 공유된 세션 키 데이터의 해쉬 값 일부를 사용하여 미리 지정된 여러 음악 샘플 파일 중 하나의 샘플 파일 재생을 통해 사용자로 하여금 빠르게 공유키를 확인 할 수 있도록 구현한 HiB 기술의 실험결과를 제시한다.

3.3 BiB 기법

디스플레이가 없는 디바이스의 경우는 변환된 공유키를 비트의 형태로 변환하여 LED로 표현하는 BiB 방법 [4]을 사용한다. 본 논문에서는 공유된 세션 키 데이터의 해쉬 값 중 8개의 비트를 일렬로 배열된 8개 LED의 점등 또는 소등 값으로 매핑하는 BiB 기술의 실험결과를 분석한다.

4. 세션 키 검증 기술 구현

본 장에서는 스마트폰을 이용한 모바일 지갑 서비스에 필요한 근거리 모바일 디바이스 간 세션 보호 기술들의 구현을 통해 실험결과를 분석한다.

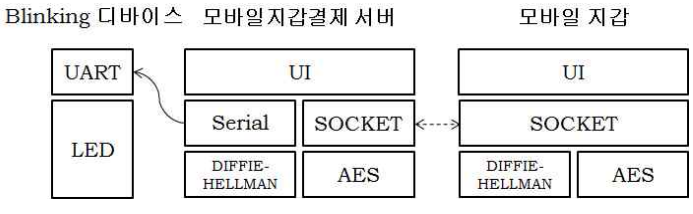
모바일 지갑과 모바일 지갑 결제 서버 사이에서 보안모드를 통한 결제정보 교환에서 보여 지는 양 단말 간 세션 키 검증을 위해 기존 SiB, HiB, BiB 기법을 적용한다.

4.1 개발환경

모바일 지갑의 경우 Windows Mobile 6.1 Professional 기반으로, 모바일 지갑 결제 서버의 경우 Windows 7 x86 기반으로 개발 하였다. BiB 기술 구현에 사용될 디바이스는 Digilent의 Spartan-3 보드를 사용하였다.

4.2 시스템 구성

모바일 지갑 결제 시나리오가 동작 하는 시스템의 구성은 (그림 4)와 같다.



(그림 4) 모바일 지갑 시스템 구조

결제 서버와 모바일 지갑 단말 시스템 구성에서 세션 키 교환을 위해 DH 프로토콜이 동작하는 모듈과 생성된 세션 키로 메시지를 암호화 하는 AES 모듈이 하위 계층에 포함된다. 두 단말은 소켓 통신을 통해 정보를 주고받는다. 또한 모바일 지갑 결제 서버에는 Blinking 디바이스의 데이터 통신을 위한 시리얼 통신 모듈이 포함된다.

4.3 시스템 동작 시나리오

모바일 지갑을 통한 상품 구매 과정에서 모바일 지갑과 모바일 지갑 결제 서버와의 키 인증 시나리오는 (그림 5)와 같다.

(그림 5) 모바일 지갑 보안모드 결제 과정

1. **키 셋업**: DH 프로토콜을 통해서 작성된 128 비트 의 세션 키를 생성한다. 이를 키 K라 한다.
2. **키 검증**: 1 단계에서 생성된 키 K를 SHA-1 알고리즘으로 160 비트 해쉬 값을 생성한다. 이를 해쉬 값 S라 한다. 키 검증은 모바일 지갑 사용자와 모바일 지갑 결제 서버 사용자의 상호작용을 통해 이루어지며, 이때 앞서 기술한 SiB, HiB, BiB의 세 가지 기법을 적용한다. 세 가지 방법 중 하나의 키 검증 방법을 사용하여, 사용자가 검증한 키만이 비밀 키로써 암호화에 사용될 수 있다. 이때, 검증된 키를 V라 한다.
3. **결제정보 전송**: 검증된 128 비트 키 V를 사용하여, AES 암호화 알고리즘으로 결제정보를 암호화 한다. 암호화된 결제 정보는 모바일 지갑에서 모바일 지갑 결제 서버로 전송한다.

4.4 세션 키 검증 기술 구현

키 검증 단계에서는 전 단계에서 생성된 세션 키 K를 SHA-1 알고리즘을 이용하여 해쉬 값을 추출한다. 이 해쉬 값을 사용하여 SiB, HiB, BiB의 키 검증 방법 중 하나를 수행하게 된다.

4.4.1 SiB 구현

160 비트 해쉬 값 중 상위 128 비트의 해쉬 데이터를 사용한다. 먼저 아래와 같이 128 비트의 데이터를 2 비트씩 자른다. 2 비트씩 잘라진 총 64개의 데이터를 (그림 7)과 같이 8x8 그리드에 미리 지정된 4가지 색상을 통해 표현한다. (그림 6)은 해당 기능을 수행하는 HashToColor( ) 함수의 코드를 나타낸다.

```
private int[ ] HashToColor(byte[ ] hashedValue) {
  int[ ] colors = new int[m_ * n_];
   for (int i = 0; i < m_ * n_ / 4; i++) {
      if( i < hashedValue.Length){
       colors[i*4] = (hashedValue[i] & 0xC0) >> 6;
       colors[i*4+1] = (hashedValue[i] & 0x30) >> 4;
       colors[i*4+2] = (hashedValue[i] & 0x0C) >> 2;
       colors[i*4+3] = (hashedValue[i] & 0x03);
      }
      else{
         colors[i*4] = 0;
         colors[i*4+1] = 0;
         colors[i*4+2] = 0;
         colors[i*4+3] = 0;
      }
   } return colors;
}
```

(그림 6) SiB 기법 구현 코드

(그림 7) SiB 2차원 바코드

4.4.2 HiB 구현

160 비트의 해쉬 값을 모두 더한 후 10으로 모듈러 연산을 수행하고, 미리 정의된 10개의 wav 파일에 각각 매핑하여 수행하게 된다. (그림 8)은 해당 기능을 수행하는 코드를 나타낸다. 세션 키에 따라 매핑된 wav 파일은 (그림 9)와 같은 화면 아래에서 재생된다.

```
for (int i = 0; i < Data.Instance.hashedKey.Length;
i++)
song += Data.Instance.hashedKey[i];
song %= 10;
String strAppDir = System.IO.Path.GetDirectoryName(
System.Reflection.Assembly.GetExecutingAssembly().
GetName().CodeBase);
string filePath = System.IO.Path.Combine(strAppDir,
"Waves\\" + song.ToString() + ".wav");
simpleSound_ = new SoundPlayer(filePath);
simpleSound_.Play();
```

(그림 8) HiB 기법 구현 코드

(그림 9) HiB 동작 화면

4.4.3 BiB 구현

해쉬 값 160 비트 중 상위 두 번째 바이트를 읽어와 해당 바이트의 최하위 비트부터 1 비트씩 추출한다.

(그림 10)은 해당 기능을 수행하는 코드를 나타낸다. 추출한 값은 (그림 11)과 같이 1은 붉은색, 0은 흰색으로 표현된다. 붉은색은 LED 의 On 상태, 흰색은 LED의 Off 상태를 의미한다.

Blinking 디바이스는 시리얼 통신을 통해 한 바이트 데이터를 전송한다. 입력받은 한 바이트의 8개 비트를 8개의 각 LED의 값으로 사용한다. 입력받은 한 바이트의 비트열 순서 그대로 LED값을 나타내는 1 바이트 크기의 변수에 대입 시킨다. LED는 0에서 7번 비트까지를 각각 순서대로 표현하며, 각 비트의 1값은 점등 0값은 소등을 나타낸다.

5. 성능 분석

사람에 의해 직접 세션 키 인증을 수행하는 SiB, HiB, BiB 의 기술의 성능을 비교 분석하면 (표 1)과 같다. 각 성능 평가 범주를 직관성, 동기화, 범용성의 세 부분으로 나누어 비교 평가 한다.

```
blinkingStyle_ = blinkingStyle;
g_ = CreateGraphics();
led = new int[8];
byte mask = 0x01;
for(int i = 7; i >= 0 ; i--){
    if( (mask & blinkingStyle_) > 0)
          led[i] = 1;
       else
          led[i] = 0;
       mask *= 2;
   }
blinkCnt_ = 2;
blinking_ = new object();
```

(그림 10) BiB 기법 구현 코드

(그림 11) BiB 동작 화면

(표 1) 세션 키 근거리 검증 기술 성능 비교 평가

	SiB		HiB		BiB	
	바코드	그리드	단어	음악	Single LED	8 LED
직관성	미흡	양호	양호	우수	미흡	양호
동기화	불필요	불필요	필요	불필요	필요	불필요
범용성	우수	양호	우수	양호	우수	양호

5.1 SiB 분석

기존의 제안된 SiB 기술은 바코드 형태로 모바일 디바이스 화면에 나타낸다. 하지만 본 논문에서 제안된 구현 방법은 다양한 색상을 이용하여 그리드 형태로 디스플레

이 해준다. 이 방법은 기존의 바코드 형태 보다 가독성이 뛰어나 직관성을 높여주고, 추후 비디오카메라를 통한 검증 방법을 사용함에 있어서도 바코드 형태보다 더욱 높은 인식률을 보여줄 수 있다. 하지만 흑백의 색상으로 이루어지는 바코드 형태에 비해 여러 색상이 필요한 그리드 형태는 디바이스의 디스플레이 성능 요구사항이 더 높다는 단점이 존재한다.

5.2 BiB 분석

BiB 기술은 SiB 기술을 적용할 수 있는 디스플레이가 없는 디바이스의 경우 LED로 키 검증을 할 수 있도록 한 기술이다 [4]. 기존의 제안된 BiB 기술 중 Single LED를 사용한 방법 [4]은 한 개의 LED 점멸 패턴 분석을 통해 세션 키 검증을 수행한다. 이에 반해 본 논문에서 소개한 BiB기술은 8개의 LED를 사용하여 점멸 패턴을 통한 검증이 아닌 LED 점등 포지션 분석을 통해 세션 키 검증을 수행한다. LED 점멸 패턴을 통한 검증 기술은 양 디바이스 LED 점멸의 동기화를 필요로 하지만 LED 점등 포지션을 통한 검증 방법은 동기화가 필요 없다는 장점이 있다. 또한 점멸 패턴의 미세한 차이로 인해 직관성이 떨어질 수 있는 부분도 LED 점등 포지션을 통한 검증 방법에서는 발생하지 않는다. 하지만 Single LED에 비해 디바이스에 여러 개의 LED를 필요로 한다는 단점이 존재한다.

5.3 HiB 분석

기존의 제안된 HiB 기술은 여러 주어진 단어 재생 순서에 따른 검증 방법을 사용하지만 본 논문에서 제안된 HiB 기술은 세션 키에 따른 음악을 선택 재생하게 된다. 사람의 감각기관을 통해 직접 키 검증을 수행한다고 했을 때, 단어의 순차적 재생은 여러 단어의 재생순서를 마지막 단어까지 들어봐야 한다. 또한 사람의 기억력 문제에 따라 두 디바이스의 단어 음성 재생은 동기화 되어야만 효과적인 세션 키 검증을 이룰 수 있다. 반면, 음악 재생을 사용하는 HiB 기술은 음악의 특정 한 부분만을 짧게 재생 해 줌으로써 단어 음성 재생 기술에 비해 키 검증 시간이 짧고 두 디바이스간의 정밀한 동기화가 필요 없다는 장점이 있다. 하지만 비교적 적은 수의 단어 순서 조합만으로 키 검증이 가능한 단어 기술에 비해 많은 음악 샘플 파일 저장 공간을 필요로 하는 공간적 비용이 큰 것이 단점이다.

6. 결론

모바일 지갑과 근거리 무선 디바이스 사이의 안전한 페어링을 위해 세션 키 교환 방법으로 간단하지만 효과적으로 키 분배를 수행할 수 있는 DH 프로토콜을 사용하고, 해당 프로토콜의 취약점인 중간자 공격을 방지 할 수 있는 기술로 SiB, HiB, BiB 기법을 통한 공유키 검증 기술의 성능을 실험을 통해 살펴보았다. SiB, HiB, BiB 기술 중 어느 한 기술이 가장 효과적인 기술이라 판단 및 적용하기 보다는, 사용되는 모바일 디바이스의 하드웨어적 제약 사항과 응용환경에 따라 선택적으로 적용하는 것이 바람직해 보인다.

본 논문에서 해당 기술들은 모바일지갑 결제 시나리오를 바탕으로, 모바일지갑 결제 시스템을 통해 구현하였다. 기존 SiB 기술이 카메라 인식을 통한 검증 방법을 제안한 반면 본 논문에서는 직접 사람이 물리적으로 세션 키의 동일성 검증을 수행하도록 제안하였다. 사람의 인지능력을 바탕으로 한 물리적 검증 방법은 비디오 카메라나 마이크와 같은 2차적 디바이스 모듈을 통해 검증하는 방법에 비해 추가적인 비용을 발생 시키지 않는다는 점에서는 효과적이나, 사람의 시각적 인지능력이 완벽하지 않을 수 있다는 점에서 정밀한 검증이 되지 못할 수 있다는 약점이 존재한다. 따라서 앞으로 본 논문에서 구현한 SiB, HiB, BiB 기술을 효과적으로 가공하여 정확한 검증 할 수 있는 2차적 인증 채널 프레임워크 설계에 대한 연구가 필요하다.

참고문헌

[1] Behrouz A. Forouzan, "Cryptography and Network Security", McGraw-Hill, 2008

[2] Jonathan M. McCune, Adrian Perrig, and Michael K. Reiter, "Seeing-is-believing: Using Camera Phones for Human-verifiable Authentication", IEEE Symposium on Security and Privacy, pp. 110-124, 2005

[3] Michael T. Goodrich, Michael Sirvianos, John Solis, Gene Tsudik, and Ersin Uzun, "Loud and Clear: Human-Verfiable Authentication Based on Audio", IEEE International Conference on Distributed Computing Systems(ICDCS'06), pp. 10-10, 2006

[4] Nitesh Saxena, "Secure Device Paring based on a Visual Channel", IEEE Symposium on Security and Privacy, pp. 306-313, 2006

[5] “사용자 중심 ID 관리 기능을 제공하는 전자ID지갑 시스템”, 전자통신동향분석, 2008.8.15

[6] “인터넷 개인 정보 유출과 전자ID지갑”, 주간기술동향, 2008.8.31

[7] 한국전자통신연구원, “Digital Identity Management - 2008년 기술 백서”, 2008.11