2000 년 한국정보처리학회 추계 학술발표논문집 제 7 권 제 2 호

이동 통신 환경에서의 키 복구 프로토콜

이병래, 장경아, 김태윤
고려대학교 컴퓨터학과
e-mail : brlee@netlab.korea.ac.kr

Key Recovery Protocol for Mobile Communication Systems

Byung-Rae Lee, Kyung-Ah Chang, Tai-Yun Kim
Dept. of Computer Science & Engineering, Korea University.

요 약

본 연구에서는 UMTS[1]에서의 보안 서비스를 실험 하고 검증하기 위한 ASPeCT[2]에서 제시한 AIP[3,4] 프로토콜을 위한 개선된 키 복구 프로토콜을 제안한다. AIP 프로토콜은 사용자와 VASP 간에 이루어지는 인증과 지불 초기화를 위한 프로토콜이며 이를 기반으로 하여 이동 통신 기반의 전자상거래가 실현될 수 있다. 제안된 키 복구 프로토콜은 검증 가능한 비밀 공유 프로토콜[5,6,7]을 기반으로 하여 다수의 키 복구 기관과의 연관을 통하여 개선된 안전성을 제공하고 악의적인 사용자가 잘못된 비밀 정보를 키 복구 기관(KRA)에게 위탁하는 것을 방지한다.

1. 서론

제 3 세대 이동 통신 시스템인 UMTS (Universal Mobile Telecommunication System)[1] 환경에서의 사용자(User)와 VASP(Value-Added Service Provider)간에 보안성을 제공하기 위하여 ASPeCT (Advanced Security for Personal Communication Technology)[2]에서는 AIP(Authentication and Initialization of Payment) 프로토콜을 제시하고 있다. 본 논문에서는 제 3 세대 이동 통신 시스템인 UMTS 에서의 AIP 프로토콜을 위한 키 복구 프로토콜을 제안한다. 제안한 프로토콜은 키 복구를 위하여 사용자의 비밀 정보를 키 복구 기관(KRA: Key Recovery Agent)에게 위탁할 경우 비밀 공유 기법을 통하여 다수의 신뢰 기관과 연관을 가지는 기법으로 개선된 안전성을 제공할 수 있다. 제안한 기법에서는 키 복구를 위하여 사용자가 자신의 초기 비밀 정보를 KRA 에게 등록 시에 거짓 정보를 제공하는 것을 방지하기 위하여 검증 가능한 비밀 공유 기법을 기반으로 하였다.

본 논문의 구성은 다음과 같다. 2 장에서는 이동 통신 환경에서의 ASPeCT 와 AIP 프로토콜에 대하여 설명한다. 3 장에서는 AIP 프로토콜에서의 키 복구 프로토콜에 대하여 기술한다. 4 장에서는 비밀 공유 기법을 기반으로 한 새로운 키 복구 프로토콜을 제안한다. 5 장에서는 제안한 프로토콜에 대한 분석이 있으며 마지막으로 6 장에서는 결론을 제시한다.

2. 이동 통신 환경에서의 보안

본 장에서는 UMTS 환경에서의 보안성을 제공하기 위한 ASPeCT 에 대하여 고찰한다. 2.1 절에서는 ASPeCT 의 배경과 공개키 기반 구조에 대하여 설명하며 2.2 절에서는 ASPeCT AIP 프로토콜에 대하여 기술한다.

2.1 ASPeCT

제 2 세대 시스템의 경우에 있어서 UMTS 를 위한 가장 기본적인 보안 요소는 기존의 유선 망과 대등한 수준의 보안을 제공하는 것이다. 이 같은 요구에 대응하는 요소들은 다음과 같다. 무선 인터페이스에서의 기밀성, 사용자 신원의 익명성 그리고 가장 중요한 것은 사용자의 네트워크로의 인증이다.

이 같은 특징들이 이미 존재하는 제 2 세대 시스템에서 제공되고 있지만 UMTS 로 발전하기 위해서는 더욱 개선된 요소들이 필요하다. UMTS 에 있어서 가장 중요한 보안 요소는 사용자는 침입자가 네트워크 오퍼레이터(network operator) 또는 서비스 제공자(service provider)를 가장하는 것을 방지하기 위하여 반드시 네트워크를 인증하여야 한다는 것이다[2].

이 같은 요소는 사용자가 자신이 신뢰하는 네트워크에 접속하고 있다는 사실의 확인을 원할 수 있다는 데에 기인한다. 이것은 공개 또는 사설 네트워크 오퍼레이터 또는 서비스 제공자가 많이 늘어나고 있어서 점점 중요해지고 있다.

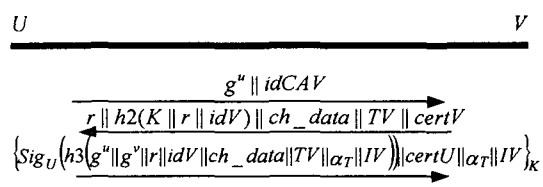
많은 분야에서의 성공적인 결과에도 불구하고 공개키 암호 시스템은 성능상의 문제로 인하여 이동 통신 시스템에서는 사용이 제약되어 왔다. 공개키 암호 시스템은 메시지의 길이와 계산량에 있어서 2 세대 시스템에 있어서는 적합하지 않았다 UMTS 환경에서의 보안 서비스를 실험하고 검증하기 위한 ASPeCT 에서는 기존의 이동 통신 시스템에서는 적용이 어려웠던 공개키 암호 시스템[8,9]을 도입함으로써 개선된 암호 프로토콜을 제시하고 있다. 이는 타원 곡선 공개키 암호 시스템을 이용하여 적은 비트 수와 빠른 계산 속도를 보장 받을 수 있는 데에 있다.

2.2 AIP 프로토콜

ASPeCT 에서의 AIP 프로토콜[3,4]은 이동 통신 환경에서의 전자상거래를 가능하게 해주는 인증과 지불 초기 프로토콜이다. AIP 프로토콜에서는 온라인 TTP 가 참여 여부에 따라서 두 가지 종류의 프로토콜로 구분되어 질 수 있다.

AIP 프로토콜의 참여자는 사용자(U)와 VASP(V)로 이루어진다. U 와 V 는 서로간의 신원을 인증하고 Diffie-Hellman 기법을 사용하여 비밀 세션 키를 성립한다. AIP 프로토콜은 이산 대수 문제(Discrete Logarithm Problem)가 어렵다는 가정을 두고 있으며 큰 소수 p 와 위수가 $p-1$ 인 곱셈상의 Z_p^* 군의 생성자 g 가 요구된다.

$idCAV$ 는 V 의 신뢰기관의 신원을 의미하며, $certX$ 는 X 의 인증서를 뜻한다. U 의 메시지 M 에 대한 전자서명 알고리즘은 각각 $Sig_U(M)$로 표기된다. 세션키 K 로 암호화된 메시지 M 은 $\{M\}_K$ 로 나타내어진다.



<그림 1> AIP 프로토콜

프로토콜(<그림 4>)이 시작되면 U 는 난수 u 를 생성하고 ElGamal 공개키 g^u 를 계산하여 $idCAV$ 와 같이 V 에게 보낸다.

V 는 난수 r 을 생성하고 세션키 $K = h1((g^u)^v \| r)$ 을 계산한다. V 는 자신이 세션키 K 를 가지고 있다는 정보를 해쉬 값 $h2(K \| r \| idV)$ 를 계산하여 인증서 $certV$, r 그리고 지불 정보에 따르는 부가 정보들을 타임스탬프 TV 와 같이 전송한다.

U 는 세션키 $K = h1((g^u)^v \| r)$ 를 계산한다. 그리고 해쉬 값 $h2(K \| r \| idV)$ 을 검사하여 V 가 실제로 세션키 K 를 가지고 있는지를 확인한다.

마지막 메시지를 받은 V 는 세션키 K 를 이용하여 복호화하여 $certU$ 를 얻고 이를 이용하여 사용자의 서명을 검증할 수 있다.

3. 이동 통신 환경에서의 키 복구 프로토콜

본 장에서는 기존의 키 복구 프로토콜을 살펴보고 그에 대한 개선점을 제시한다. 3.1 절에서는 제 3 세대 이동 통신 환경을 위한 키 복구 프로토콜[10,11]을 살펴보고 3.2 절에서는 키 복구 프로토콜의 개선된 안전성을 보장하기 위한 다수의 KRA 와의 연동 방법을 설명한다.

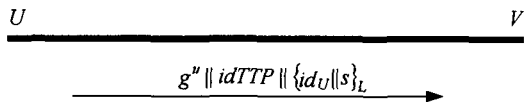
3.1 기존의 키 복구 프로토콜

[10]에서는 ASPeCT AIP 프로토콜에서 U 와 V 간에 생성되는 비밀 세션 키를 복구하기 위한 방법들을 제안하고 있다. 기본적인 개념은 다음과 같다. KRA(Key Recovery Agent)는 키 복구를 하는 기관으로서 TTP 가 담당할 수 있다.

키 복구 기법은 u 가 생성되는 방법을 수정함으로써 얻어질 수 있다. 즉, u 는 다음과 같이 계산된다.

$$u = f(w, s)$$

여기서 f 는 일방향 함수이고 s 는 일회용 난수 초기값이며 w 는 U 와 KRA_U 간의 등록 단계에 미리 설정된 비밀 정보이다. 아래의 메시지는 U 와 U 의 KRA 와의 세션 키 L 로 암호화된 U 의 신원 id_U 와 비밀 정보 s 를 가지고 있다.



<그림 2> 키 복구 프로토콜

U 의 도메인에서 키 복구 과정은 다음과 같이 진행된다. 키 복구를 요구하는 엔티티는 KRA_U 역할을 하는 U 의 TTP 에게 일회용 난수 g^u , V 의 인증서 $certV$, 난수 r 과 암호화된 $\{id_U \| s\}_L$ 을 전송한다. KRA_U 는 $\{id_U \| s\}_L$ 을 복호화 하여 id_U 을 보고 w 를 얻는다. w 와 s 를 이용하여 KRA_U 는 u 를 계산할 수 있고 r , g^v 를 이용하여 세션키 $K = h1(g^{uv} \| r)$ 를 계산해 낸다. 그러나 V 의 도메인의 경우에 있어서는 키 복구 과정이 다르다. V 는 자신의 TTP 에게 비밀 키 설정 키 v 를 등록 해야 한다. V 의 TTP 는 V 의 비밀키 v 와 g^u 를 이용해서 세션키 K 를 계산해 낼 수

있다.

[11]에서는 [10]의 단점과 이를 해결한 프로토콜을 나타내고 있다. 또한 [11]에서는 U 가 고의적으로 잘못된 비밀 분할을 KRA_U 에게 전달하는 것을 방지하기 위하여 아래와 같은 방식으로 세션키를 생성해 낸다.

$$K = f(r, \text{other public information})$$

f 는 공개적으로 알려진 일방향 함수이고 r 은 사용자로부터 생성된 비밀 난수이다. 공개적으로 검증 가능한 방법을 사용하여 올바른 r 값인지에 대한 결과를 얻을 수 있게 하였다. 이 경우에 있어서 키 복구는 r 을 복구 하는 것과 유사하다. 이와 같은 방법은 U 의 도메인에서는 키 복구가 가능하나 V 의 도메인에 있어서는 가능하지 않은 단점이 있다.

[11]에서는 개선된 안전성을 얻기 위하여 U 의 정보 w 를 다수의 KRA 와 공유하는 방식을 취하고 있다. 그러나 다수의 KRA 와 연합하는 기법의 경우 U 가 정확한 초기 비밀 정보 w_i 를 제공하였는지에 대한 검증 방법이 제공되어 있지 않다.

3.2 비밀 공유와 키 복구 프로토콜

키 복구를 가능하게 하기 위하여 사용자는 초기에 자신의 신뢰 기관에게 비밀 정보를 제공한다. 그러나 사용자가 고의적으로 키 복구 과정을 실패하도록 하기 위하여 잘못된 비밀 정보를 제공할 수 있는 문제점이 있다. 이를 방지하기 위하여 []에서는 사용자가 제공한 비밀 정보를 검증할 수 있는 기법을 나타내었다. 그러나 이 기법은 비밀 정보를 여러 개로 분할하여 다수의 신뢰 기관과의 프로토콜 진행 시에는 적용할 수 없다.

본 연구에서는 사용자가 자신의 비밀 정보를 신뢰 기관들에게 전송 시에 다수의 신뢰 기관들이 자신이 받은 비밀 정보 분할이 올바른 것인가를 검증 할 수 있는 기법을 제안한다. 이를 위하여 검증 가능한 비밀 공유 기법을 키 복구 등록 프로토콜에 적용하여 각 신뢰 기관들이 사용자로부터 제공받은 비밀 분할을 검증할 수 있도록 하였다.

4. 검증 가능한 비밀 공유 기반 키 복구 프로토콜

본 장에서는 검증 가능한 비밀 공유 기법을 키 복구 등록 프로토콜에 적용하여 사용자의 비밀 분할을 제공받는 신뢰 기관이 검증 할 수 있는 개선된 안정성을 보장하는 방법을 제안한다. 4.1 절에서는 검증 가능한 비밀 공유 기법을 설명하고 4.2 절에서는 제안한 키 복구 모델에 대하여 간략히 나타낸다. 4.3 절에서는 AIP 프로토콜을 기반으로 하여 이동 통신 환경에서의 검증 가능한 비밀 공유 기반 키 복구 프로토콜을 제안한다.

4.1 검증 가능한 비밀 공유 프로토콜

Shamir[5]는 임계치 기법의 구성을 위해 유한체의 다항식을 사용하였다. (k,n) 임계치 비밀 공유 기법은 n 참여자들에게 비밀에 관한 부분 정보를 분배하는 딜러(dealer)가 있는 n 참여자들 사이의 프로토콜이다. 이러한 비밀 공유 기법은 k 보다 작은 참여자들의 그룹이 비밀에 관한 어떠한 정보도 얻을 수 없으며, 적어도 k 참여자들의 그룹은 다항 시간(polynomial time)에 비밀을 계산할 수 있다.

Shamir 의 비밀 공유 방식은 모든 참여자들이 정직하다는 가정 하에서 유효한 프로토콜이다. 하지만 실제 응용에서는 악의 있는 적을 고려할 필요가 있다. 공유된 비밀의 정당성을 검증할 수 있는 프로토콜을 VSS(Verifiable Secret Sharing)이라고 한다. 대표적인 VSS 프로토콜로는 Feldman[6]과 Perdesen[7]의 것들이 존재한다.

본 논문에서는 검증 가능한 비밀 공유 프로토콜을 이동 통신 환경에 적용하기 위하여 다음과 같은 세가지의 프로토콜들로 표현한다. 아래의 비밀 공유 프로토콜에서는 전체 n 개의 참여자가 있다고 가정하였다.

Share : 딜러는 n 개의 참여자들에게 비밀을 검증 가능한 방식으로 분할하여 안전하게 전송한다. 비밀 s 는 ***Share*** 프로토콜을 통하여 $s_1, \cdots, s_n$ 의 비밀 분할로 변형된다.

$$Share(s) = (s_1, \cdots, s_n)$$

Recover : 전체 n 개의 참여자중 적어도 k 개의 참여자들이 모이면 비밀 s 를 재구성 할 수 있다. 여기서의 A 는 비밀 공유 기법에서의 접근 구조(Access Structure)이다. 단 $i, (0 \le i \le n)$ 이다.

$$Recover(\{s_i \mid i \in A\}) = s$$

Verify : n 참여자들은 자신의 비밀 분할이 올바른것인지에 대해 검증한다.

$$Verify(s_i) = true \text{ or } false$$

4.2 제안한 비밀 공유 기반 키 복구 프로토콜

제안한 키 복구 모델은 다음과 같이 이루어 진다. U 의 도메인에는 키 복구 기관 KRA_{Ui} $(1 \le i \le n)$ 가 있으며 V 의 도메인에는 키 복구 기관 KRA_{Vj} $(1 \le j \le m)$ 가 존재한다. 제안된 키 복구 모델은 [10]에서 제시된 AIP 프로토콜의 기본적인 키 복구 모델을 기반으로 하여 U 의 초기 비밀 정보 w 와 V 의 비밀키 v 를 각각의 KRA 에게 위탁하는 방식을 이용하였다.

등록 단계 : 등록 프로토콜은 U 와 V 가 AIP 프로토콜을 수행 전에 자신들의 KRA 인 TTP 와 시행이 된다.

U 는 자신의 초기 비밀 정보 w 를 ***Share*** 프로토콜을 이용하여 n 개의 KRA_{Ui} 에게 안전하게 전송한다.

V 도 유사한 방법으로 자신의 비밀키 v 를 ***Share*** 프로토콜을 이용하여 m 개의 KRA_{Vj} 에게 안전하게 전송한다.

등록 단계 종료 후 U 와 V 는 KRA 역할을 하는 자신의 TTP 들과 검증 가능한 비밀 공유 기법을 통해 초기 비밀 정보 w 와 v 를 공유하게 된다.

검증 단계 : 검증 단계에서는 각 TTP 들은 U 와 V 로부터 전송 받은 w_i 와 v_j 가 올바른것인지에 대하여

각 KRA_{Ui} 는 자신이 제공받은 비밀 분할 w_i 이 올바른 것인지에 대하여 검증한다. 만약 ***Verify*** 프로토콜을 통한 검증이 실패하면 검증을 한다. 검증이 실패하면 KRA 는 다른 KRA 들에게 브로드캐스트하고 .프로토콜의 수행은 정지된다.

KRA_{Vj} 도 자신이 제공 받은 비밀 분할 vi 을 ***Verify*** 프로토콜을 통하여 검증한다. 만약 검증이 실패하면 검증을 한다. 검증이 실패하면 KRA 는 다른 KRA 들에게 브로드캐스트하고 프로토콜의 수행은 정지된다.

검증 단계까지 정상적으로 완료가 되면 U 와 V 는 AIP 프로토콜을 수행할 수 있다.

키 복구 단계 : U 와 V 간의 통신 내용을 살펴보고 싶은 기관은 KRA 에게 키 복구를 요청할 수 있다.

U 의 도메인의 KRA_{Ui} 는 키 복구 요청을 받아서 ***Recover*** 를 통하여 비밀 w 을 계산해내고 U 와 V 간에 비밀 세션 키 K 를 복구 한다.

V 의 의 도메인의 KRA_{Vj} 는 키 복구 요청을 받아서 먼저 ***Recover*** 프로토콜을 통하여 비밀 v 을 재조립한 후에 U 와 V 간에 비밀 세션 키 K 를 복구 할 수 있다.

4.3 비밀 공유 기반 키 복구 프로토콜 고찰

4.2 절에서 제안한 비밀 공유 기반 키 복구 프로토콜은 등록 단계 시에 사용자와 KRA 간에 비밀 공유가 이루어지므로 실제 AIP 프로토콜 수행 시에는 별다른 계산량의 증가가 없다.

사용자의 계산 능력을 고려할 때에는 사용자의 이동 단말이 직접 비밀을 분할하여 KRA 와 공유하는 방식을 취할 수도 있지만 KRA 들이 비밀을 생성하여 사용자에게 제공할 수도 있다.

5. 제안한 키 복구 프로토콜 분석 및 평가

제안한 비밀 공유 기반 키 복구 프로토콜은 UMTS 환경을 위한 기존의 키 복구 프로토콜에 비하여 다음과 같은 두 가지의 특징을 가진다.

- 비밀 공유 기법을 적용하여 사용자의 비밀 정보를 다수의 KRA 에게 공유함으로써 개선된 안전성을 제공한다.
- 다수의 KRA 가 자신이 사용자에게 제공 받은 비밀 정보가 올바른 것인지에 대한 검증을 할 수 있어 사용자의 고의적인 행위를 방지할 수 있다.

그러나 사용자가 자신의 비밀 정보를 공유하기위한 초기 계산량이 다른 방법에 비하여 크다는 단점이 있을 수 있다.

6. 결론 및 향후 연구 과제

본 논문에서는 검증 가능한 비밀 공유 기법에 기반하여 이동 통신 환경에서의 키 복구 프로토콜을 제안하였다. 제안된 키 복구 프로토콜은 사용자와 신뢰 기관의 초기 비밀 정보 등록 시에 비밀 공유 기법을 기반으로 하여 신뢰 기관들에게 사용자의 비밀 분할을 제공하였다. 또한 사용자가 고의적으로 잘못된 분할을 재시하는 것을 방지하기 위하여 비밀 분할을 검증 가능하도록 하였다.

향후 연구 과제로는 사용자와 KRA 가 공유하는 비밀 정보의 갱신이 주기적으로 이루어지도록 하여 개선된 안전성을 보장하는 것이다[12].

참고문헌

[1] UMTS Forum, "A regulatory framework for UMTS," Report no. 1, 1997.

[2] ACTS AC095, ASPeCT Deliverable D20 – Project final report and results of trials, 1998.

[3] G. Horn and B. Preneel, "Authentication and payment in future mobile systems," *ESORICS*, *LNCS*, vol.1488, *pp.* 469-472 1998.

[4] K.M. Martin, B. Preneel, C. Mitchell, H.J. Hitz, G. Horn, A. Poliakova and P. Howard, "Secure billing for mobile information services in UMTS," *IS&98*, *LNCS* vol.1430, pp. 535-548, 1998.

[5] A. Shamir, "How to share a secret," Communication of the ACM, Vol.22 pp.612-613, 1979.

[6] P. Feldman, "A Practical Scheme for Non-Interactive Verifiable Secret Sharing," *IEEE Proc. 28th FOCS*, pp.427-437, 1987.

[7] T. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. *In Advances Cryptology-Crypto '91*, *LNCS* vol.547 pp.129-140, 1991.

[8] W. Diffie and M. Hellman, "New Directions in Cryptography," *IEEE Transactions on Information Theory*, 1976.

[9] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Informatio Theory*, Vol. IT-31, No.4, pp.469-472, 1985.

[10] Rantos K. and Mitchell C., Key recovery in ASPeCT authentication and Initialisation of payment protocol, *Proceedings of ACTS Mobile Summit*, Sorrento, Italy, June 1999.

[11] J. G. Nieto, D. Park, C. Boyd and E. Dawson, "Key Recovery in Third Generation Wireless Communication Systems," *PKC*, *LNCS* vol.1751, pp.223-237, 2000.

[12] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung. Proactive Secret Sharing or: How to cope with perpetual leakage. *In Advances Cryptology – Crypto '95. LNCS* vol. 963 pp.339-352, 1995.