분배된 키위탁 시스템을 위한 확률적 키정보 복구

오흥룡*, 심현정*, 류종호**, 염흥열*

*순천향대학교 정보보호학과, **정보기술공학부

Probable Information-revealing System for the Distributed Key Escrow Scheme

Heung-Ryong Oh*, Hyun-Jung Sim*, Jong-Ho Ryu**, Heung-Youl, Youm*

*Department of Information Security, **Department of Information Technology Engineering, Soonchunhyang Univ.

요 약

본 논문에서는 두 참여자간의 암호화 통신상에 있어 법집행기관이 정해진 확률 P에 따라 메시지를 복호화 하는 것이 가능하도록 해주는 키위탁 기법을 고려한다. 이것은 확률 P에 의해 사람들 개개인의 프라이버시와 법집행기간에 의한 개인침해란 두 주제 사이를 적절히 조절하도록 해준다. 제안된 방법은 ElGamal의 공개키 암호, 공개적으로 검증가능한 ElGamal 공개키 암호에 있어서 공통 지수부에 대한 지식 증명, 그리고 분배된 키위탁 기관들에서의 비밀공유기법 등에 바탕을 둔다. 또한 두 참여자간의 세션키를 복호화 하기 위해 필요한 각 키위탁 기관의 파라메터들은 영지식 대화형 증명 프로토콜을 통과하여야만 사용이 가능하도록 구성된다. 이와 같은 기법들을 통해 두 참여자간의 암호화된 통신에 접근 가능한 법집행기관과 암호화된 통신에 사용된 세션키를 복호하기 위한 분배된 키위탁기관들이 참여된 키위탁 기법을 고려한다.

1. 개요

키위탁(Key Escrow) 기법은 개인의 사생활 보장과 법집행에 필요한 공권력 사이에서 오랜 동안 토론되어 왔던 이슈이다. 본 논문에서는 비대화형 분수형(fractional) 불확정 전송(OT : Oblivious Transfer)을 바탕으로 공개적으로 검증가능한 부분 정보-누설 키위탁 기법을 제안한다. 이 기법은 법집행 기관이 각 메시지마다 확률(probability) $P(0\leq P\leq 1)$로 액세스하는 것이 가능하며, 이 확률은 개개인 마다의 프라이버시와 분배된 키위탁 기관들(DEA : Distributed Key Escrow Authorities)에 의한 키복구 사이에 적절한 균형을 유지하도록 해주는 선택 값이 된다.

위와 같은 시스템의 예로 임의의 참여자 A가 다른 참여자 B에게 암호화된 메시지를 전송한다라고 가정할 경우, 분배된 키위탁 기관들은 사전에 규정된 확률 P에 따라 메시지를 복호화 하는 것이 가능하게 된다. 그러나 이 시스템은 법집행기관이 확률 P 내에서는 모든 메시지를 들여다 볼 수 있는 것이 가능하다. 이와 같은 단일 키위탁 기관의 독단적 법집행을 막기 위해 믿을 수 있는 제3자 즉 시민연합이나 비정부기관 및 다수의 공공 법률집행 기관들로 구성된 분배된 키위탁 기법을 구성한다. 이에 대한 기본 개념은 [2]에 설명되어 있다.

상대방과 안전한 통신을 원하는 참여자 A(이후 A)가 랜덤하게 세션키 S을 선택한 후 참여자 B(이후 B)에게 암호화된 메시지를 전송한다라고 가정할 경우, 법집행기관은 이를 지속적으로 도청하여 한다. 단 A는 법집행기관의 공개키를 반드시 이용하여야 하며 이에 대한 증명도 수행하여야 한다. 이후 법집행기관은 두 통신 참여자가 불법적 행위를 하고 있다고 판단되는 경우에 분배된 키위탁 기법에 따라 확률 P 만큼 세션키 S을 구할 수 있게 된다. 그러나 이점은 상당히 큰 불법적 행위라도 법집행기관이 확률 $1-P$에 해당한 세션키 S는 복구할 수 없다는 단점을 지니다. 따라서 본 논문에서는 [5]에서 제안된 방법을 통해 어떠한 세션키라도 부분적으로는 노출되며 또한 노출된 정보 역시 다수의 기관들이 동의하여야만 얻을 수 있게 된다. 이를 위해 제안된 방법의 기본 암호 기술은 ElGamal의 공개키 암호, 공개적으로 검증가능한 ElGamal 공개키 암호에 있어서 공통 지수부에 대한 지식 증명, 그리고 분배된 키위탁 기관들에서의 비밀공유기법 등에 바탕을 둔다. 또한 두 참여자간의 세션키를 복호화 하기 위해 필요한 각 키위탁 기관의 파라메터들은 영지식 대화형 증명 프로토콜을 통과하여야만 사용이 가능하도록 구성된다.

본 논문의 2장에서는 ElGamal의 공개키 암호, 공개적으로 검증가능한 ElGamal 공개키 암호에 있어서 공통 지수부에 대한 지식 증명, 그리고 분배된 키위탁 기관들에서의 비밀공유기법을 기술한다. 3장에서는 분수형 OT기법을 사용한 향상된 키위탁 기술이 기술하며, 마지막 4장에서 결론을 맺는다.

2. 사용된 암호기술

본 논문의 키위탁 기법은 기본적으로 공통지수의 지식 증명, 공개적으로 검증가능한 ElGamal 공개키 암호, 그리고 threshold ElGamal 암호를 이용한다. 이번 장에서는 이 세 가지 사항에 대하여 간략히 살펴본다. 실명에 앞서 우선 G가 공개 곱셈군(mutiplicative group)이란 설정 하에 p와 q는 $q|(p-1)$을 만족하는 큰 소수이고, $g \in Z_p^*$는 법 p에 관한 위수(order) q를 갖는 원시근이며, 밑수(base) g에 대한 이산대수를 찾는 것은 대단히 어렵다는 사항을 기본 조건으로 가정한다.

ElGamal 공개키 암호의 보안성은 위에 가정된 이산대수 문제의 어려움과 Diffie-Hellman 문제에 바탕을 둔다. ElGamal 암호를 실현하기 위해 프로토콜의 참여자($user_1$)는 큰 랜덤 소수인 법 p와 p의 곱셈군 Z_p^*에서의 생성원 g을 선택한 후, 랜덤한 정수 x $(1 \le x \le q-2)$와 $y \equiv g^x \pmod p$을 생성한다. 이때 $user_1$의 공개사항은 p, g, y이고 $user_1$의 비밀키는 x가 된다. ElGamal 공개키 암호에서 다른 참여자($user_2$)는 메시지 m을 $user_1$에게 암호화하여 보내게 되는데, 우선적으로 $user_2$는 메시지 m의 암호문 쌍 $(c_1, c_2) = (g^a, m(y)^a) \bmod p$을 계산하고 이를 $user_1$에게 전달한다. 여기에서 $a \in_R Z_q$는 임의의 정수이다. $user_1$는 수신된 암호문을 자신의 비밀키 x를 이용하여 평문 $m \equiv c_2/c_1^x$을 복호화 한다.

2.1 공통 지수의 지식에 대한 증명

이번 절에서는 주어진 두 수의 이산대수가 동일함을 증명하는 방법에 대하여 논한다. 사전조건으로 프로토콜의 증명자(prover)는 다른 검증자(verifier)에게 식 $c_1 \equiv g^a$와 $c_2 \equiv y^a$의 공통지수를 알려주지 않으면서도 자신이 알고 있음을 검증자에게 증명하길 원한다라고 하자. 이와 같은 문제의 효율적인 프로토콜은 그림 1에서와 같이 Chaum과 Pedersen [6]에 의해 논의되었다. 이 프로토콜을 통해 두 참여자는 이산대수의 동일성을 증명하고 또한 증명 받게 된다.

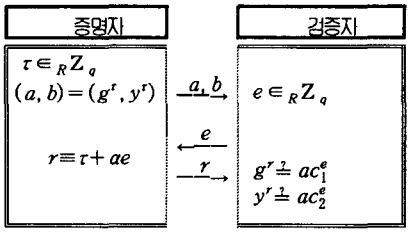

그림 1. 공통 지수의 지식에 대한 증명

2.2 공개적으로 검증가능한 ElGamal 공개키 암호에서 공통지수 지식에 대한 증명

주어진 두 수의 이산대수가 동일하다는 점을 증명하는 방법에 있어, 이 이산대수 비트열의 각 비트 마다 증명하는 영지식 증명 기법에 대하여 논한다. A는 B에게 식 $c_1 \equiv g^a$와 $c_2 \equiv y_B^a$의 공통지수 값을 보여주지 않으면서도 이를 알고 있음을 증명하기 원한다라고 하자. 여기에서 y_B은 B의 공개키이다. 이와 같은 조건에 적합한 효율적인 프로토콜은 [1]에 설명되어 있다. 본 논문에서 사용된 기본 기법은 Cramer 등이 제안한 [3]에 바탕을 두고 있으며, 또한 이산대수에 대한 Stadler의 검증가능한 암호 기법을 이용한 Wenbo Mao[4,5]의 제한기법을 본 논문에서의 키 위탁 방법으로 적합하도록 변형한 것이다.

기호 $Hash(\,)$는 Z_q에서 유일하게 정의된 일방향 함수(즉, $Hash(\,): \{0,1\}^* \rightarrow Z_q$)이고, 기호 $a \in_R Z$는 Z에서 랜덤하게 선택하였음을 의미한다. 프로토콜을 수행하기 전에 B는 자신의 비밀키 $x_B \in Z_q$를 생성함과 동시에 공개키 $y_B = g^{x_B}$을 계산한다. A가 생성한 암호문쌍 $(c_1, c_2) = (g^a, y_B^a f)$은 $f = c_2/c_1^{x_B}$과 같이 비밀키 x_B를 사용하여 복호화가 가능하다. 여기에서 $a \in Z_q$는 임의의 세션키이고, 신뢰된 제3자에 의해 선택된 f는 그룹 G의 고정된 생성원으로 누구도 $\log_\theta f$ $(\theta \in G)$을 얻는 것은 불가능하다. 그림 2에 도시된 대화형 프로토콜 C는 식 $\log_g(c_1) = \log_{y_B}(c_2/f^0)$ 또는 $\log_g(c_1) = \log_{y_B}(c_2/f^1)$ 와 같이 두 가지 경우만 존재할 때에, A가 선택한 하나의 식에 대한 이산대수 지식의 동일함을 증명하는 프로토콜이다. 여기에서 $f^0 = 1$ 와 $f^1 = f$은 1 비트 승을 의미한다.

프로토콜 수행전 각 참여자의 공통 사전 지식은 c_1, c_2, g, p, y_B, f 이다. 그림 2의 대화형 프로토콜 C를 이용하여 임의의 $a \in_R Z_q$에 대한 다음 식을 증명하게 된다. 여기에서 a는 $a = \log_{y_B}(c_2)$ 또는 $a = \log_{y_B}(c_2/f)$가 된다.

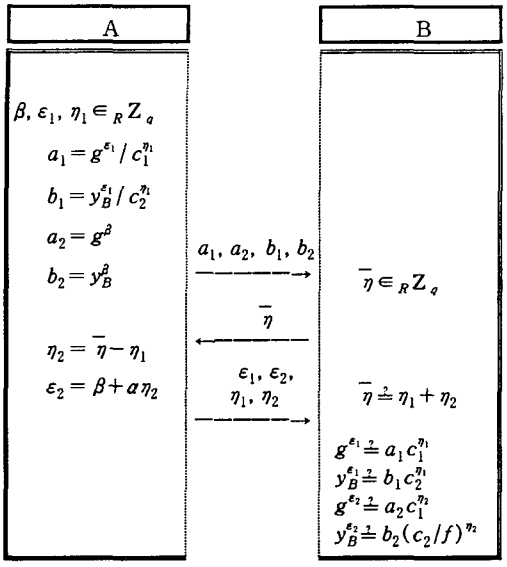

그림 2. 대화용 프로토콜 C

B는 임의의 $\alpha \in_R Z_q$ 와 f에 대한 마지막 네 가지 식을 검증한 후 프로토콜의 결과를 받아들인다. 프로토콜에서 암호문쌍 $(c_1, c_2) = (g^{\alpha}, y_B^{\alpha} f)$ 은 B에 의해 검증이 이루어지지만, 이후에 믿을 수 있는 제3의 기관에 의해 복호화가 수행되어 1 비트 정보 $f^0 = 1$ 또는 $f^1 = f$를 누설 할 수 있게 된다.

이와 같은 대화형 프로토콜은 해쉬 함수를 이용하여 비대화형으로 바꾸는 것이 가능하며 이에 대한 프로토콜이 그림 3에 도시되어 있다. 여기에서 i는 $i = 1, \cdots, n$의 범위를 갖는다.

대화형 프로토콜과 마찬가지로 프로토콜 수행 전 각 참여자의 공통지식은 c_1, c_2, g, p, y_B, f이다. 비대화형 프로토콜 C에서 A는 $a_1, a_2, b_1, b_2, \varepsilon_1, \varepsilon_2, \eta_1, \eta_2, \bar{\eta}$ 을 구성하여 B에게 전달하고, B은 마지막 네 가지 식을 n번 반복하여 검증한 후 프로토콜의 결과를 받아들인다. 최종적으로 비대화형 프로토콜 C를 이용하여 임의의 $\alpha \in_R Z_q$에 대한 $\alpha = \log_{y_B}(c_2)$ 또는 $\alpha = \log_{y_B}(c_2/f)$ 임을 증명하게 된다.

2.3 Threshold ElGamal 암호

Threshold ElGamal 공개키 암호의 목적은 일련의 수신자들 집합에게 비밀키를 분배하고, 이후 수신자들의 일부 집합이 합의할 경우에 메시지의 복호화가 가능하도록 하는 것에 있다[3,11]. 이와 같은 기법은 키생성 단계 및 복호화 단계인 두 가지 부분으로 나누어 생각해 볼 수 있다. 키생성 단계는 수신자들 모두가 서로 협력하여 비밀키를 생성하는 단계이고, 복호화 단계는 (전체 수신자 모두가 협력하여) 비밀키의 완벽한 재생성 없이도 수신자 일부가 서로 협력하여 암호문을 복호화 할 수 있는 단계이다.

1) 키생성 단계 : 수신자의 비밀키는 n명의 신뢰된 인증자들 $DEA_j (1 \le j \le n)$에게 공유되고 이후의 복호화는 t명 이상의 인증자들 연합이 이루어질 경우에만 가능하도록 구성되는 (t, n)-threshold 공개키 암호 기법을 고려한다. 각 인

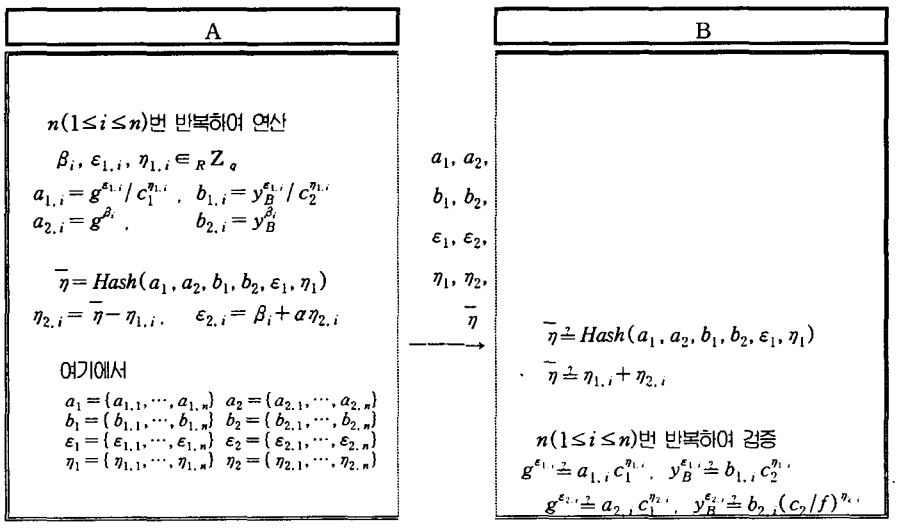

그림 3. 제안된 비대화용 프로토콜 C

증자 DEA_j는 비밀값 x의 공유값 $x_j \in Z_q$을 생성하고, 또한 이 x_j의 위탁값으로서 $y_j = g^{x_j}$을 공개한다. 비밀값 x는 라그랑주 계수법(Lagrange coefficients)을 이용하여 공유값들 x_j 중에 t개 집합인 Λ으로부터 만들어진다. 이것은 DEA의 공개키 $y = g^x \pmod p$는 t명 이상의 참여자가 동의하고 다음과 같은 식을 적용하였을 경우에만 이루어짐을 의미한다. $x = \sum_{j \in \Lambda}(x_j \lambda_{j,\Lambda})$, 여기에서 $\lambda_{j,\Lambda} = \prod_{\ell \in \Lambda \setminus \{j\}} \ell / (\ell - j)$ 이다.

2) 복호화 단계 : 공개키 y를 이용하여 메시지 m을 암호화하는 것은 식 $(c_1, c_2) = (g^a, y^a m)$으로 표현 할 수 있다. 인증자들은 비밀키 x를 재생성 하지 않고도 암호문쌍 (c_1, c_2)을 복호화하기 위해 다음과 같은 절차를 따른다.

① 각 인증자 DEA_j는 $w_j \equiv c_1^{x_j}$을 공개한 후 그림 1에 도시되이 있는 영지식 증명을 이용하여 이산대수 $\log_g(y_j) = \log_{c_1}(w_j)$의 동일성을 증명한다.

② 집합 Λ는 영지식 증명에 통과된 임의의 인증자들 집합이라 하자. 라그랑쥬 계수법을 이용하면 t명의 임의의 부분집합 Λ는 비밀키 x를 재생성하는 것이 가능하다. 즉, 평문은 식 $m = c_2 / \prod_{j \in \Lambda} w_j^{\lambda_{j,\Lambda}}$을 이용하여 구할 수 있게 된다.

3. 분수형 OT 및 ElGamal 암호를 이용한 키복구 기법

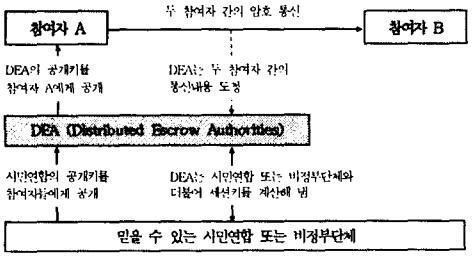

그림 4. 제안된 키복구 기법의 전체 개요

M.Bellare 과 R.Rivest는 [2]에서 두 참여자간의 암호 통신을 허가된 법 집행기관에 의해 복호화가 가능하도록 구성된 키복구 기법을 제안하였다. 이 제안된 기법은 참여자의 프라이버시 및 법 집행에 있어서의 적절한 균형을 유지하기 위한 수단으로 각 메시지에 대하여 $P(0 \le P \le 1)$ 확률로 액세스가 가능하도록 구성되며, 또한 그와 같은 기법을 제공하기 위하여 비대화형 OT에 바탕을 둔 암호 기법을 사용한다.

본 논문에서의 OT기법은 메시지 전송 확률 1/2 인 Bellare와 Micali [7]의 기법을 그림 3과 같은 비대화형 OT로 확장하여 구성한다. 또한 참여자들의 암호 통신간에 주고받는 임의의 메시지에 대하여 법 집행기관이 정해진 확률 P 만큼 메시지를 복호화 하는 것이 가능하도록 한다. 여기에서 확률 $P = a/2^w$ (여기에서, $1 \le a \le 2^w - 1$)는 임의의 유한 이진 분수 표현으로, $P = a/2^w$의 w 비트열 이진 표현은 $P = (a_1/2^1) + (a_2/2^2) + \cdots + (a_w/2^w)$와 같이 할 수 있음에 따라 $P = 0.a_1 a_2 \cdots a_w$이 된다. 기본적으로 w, a, P는 어느 누구에게나 공개되는 값들이다.

본 논문의 목적은 비대화형 분수형 OT 기법과 threshold ElGamal 암호 기법에 바탕 둔 개선된 키복구 기법을 제안하는 것에 있으며, 제안된 키 복구 기법의 전체 개요는 그림 4에 도시되어 있다. 우선 시민연합 혹은 비정부단체는 자신의 공개키를 공개하고, DEA는 이를 이용하여 여러 개의 공개키들을 생성한다. 만일 A가 B와 함께 안전한 통신을 원하는 경우, A는 DEA의 여러 개의 공개키 중 하나를 선택하여 암호화를 수행한 다음 이를 B에게 전달한다. 지속적으로 두 참여자 간의 통신을 도청한 DEA는 A가 어떤 공개키를 선택하느냐 따라 확률 P 만큼 그들의 내용을 열볼 수 있게 된다.

3.1 비대화형 분수형 OT를 이용한 시스템의 개요

OT는 Rabin에 의해 소개된 암호 기술이다. 일반적으로 OT 프로토콜은 송신자 A와 수신자 B 둘 사이에 이루어지는 프로토콜로서, A가 메시지 두 개를 B에게 전달한다라고 가정하는 경우 B는 두 개의 메시지 중에 하나의 메시지만을 선택할 수 있고 또한 A는 B가 어떤 메시지를 선택 하였지에 대하여 알지 못한다. 이는 B가 확률 0.5 만큼 메시지를 선택할 수 있고 선택되지 않은 다른 메시지를 알아내는 것은 불가능함을 의미한다. 이를 흔히 OT_2^1로 표기한다[2,7,8].

본 논문에서는 비대화형 통신 방법에 있어서 영지식 증명을 갖지 않는 비대화형 OT 프로토콜이 제안한다. 이 방법은 M. Bellare와 R. Rivest [2]에 의해 제안된 방법에 비대화형 OT_2^1을 적용하여 개선한 것이다.

1) 초기 설정 단계 : 이 단계에서 세 가지의 공개 값 p, g, Y_w을 설정한다. 이 세 가지 값은

큰 소수 p, 곱셈군 Z_q^*에서의 임의의 생성원 g, 그리고 신뢰된 비정부기관의 공개키 Y_u는 모든 참여자에게 알려진다. 단 값 $Y_u \equiv g^{x_u} \bmod p$는 믿을 수 있는 제3자인 시민단체 혹은 비정부기관에서 선택하여 발표한 값으로 어느 누구도 Y_u의 이산대수 $\log_g(Y_u)$을 알아내는 것은 불가능하다.

2) DEA에 의한 공개키 계산 단계 : 이 단계는 단 한번만 수행이 이루어진다. DEA는 $V_2 = V_1 Y_u$을 만족하는 공개키 쌍 (V_1, V_2)을 공개하며, $\log_g V_1$ 혹은 $\log_g V_2$ 값 둘 중에 하나만을 알게 있다. 따라서 앞으로 본 논문에서는 DEA가 $\log_g V_\delta$ $(\delta \in \{1,2\})$의 값을 올바르게 알고 있다면 V_δ는 "정확한 키"값이라 부르고, 그렇지 않다면 V_δ는 "부정확한 키"라 부를 것이다.

3) 통신 단계 : A가 메시지 $S \in Z_p^*$을 B에게 전달하길 원한다라고 하자. A는 DEA가 공개한 공개키 쌍 중에 한 값 V_δ을 랜덤하게 선택한 후 ElGamal 암호 알고리즘을 통해 S를 암호화한다. 이 단계에서 S를 메시지라 정의하여 이용하고 있지만, 이후에 S는 A와 B간의 암호화 통신에 사용될 세션키가 된다. A는 값 $e \in_R Z_{p-1}$을 선택한 다음 B에게 $E(S, V_\delta)$와 B의 공개키로 암호화된 $E(S, y_B)$을 전달한다. DEA가 이들 간의 통신을 도청하는 동안, 만일 DEA가 이산대수 $x = \log_g V_\delta$을 알고 있다면 DEA는 $S = c_2 / c_1^x$을 통해 S을 계산 할 수 있게 된다. 여기 $E(S, V_\delta) = (c_1, c_2) = (g^e, SV_\delta^e)$는 S가 ElGamal 공개키 암호시스템을 통해 암호화 되었음을 의미한다. 결론적으로 DEA의 두 공개키 중에 오직 하나만이 "정확한 키"이기 때문에 DEA는 확률 0.5로 S을 얻게 된다.

4) 비대화형 분수형 OT로의 확장 : 확률이 P인 비대화형 분수형 OT를 구성하기 위해, DEA는 $V_i \equiv V_0 Y_u^i$ $(i \in Z_n)$을 만족시키는 $V_0, \cdots, V_{n-1}$을 공개한다. 이들 값들 중에서 오직 하나만이 DEA의 "정확한 키"가 된다. A는 정확히 $V_i \equiv V_0 Y_u^i$을 만족하는지에 대하여 검사해 본 다음, 이들 DEA의 공개키들 중에서 하나만을 랜덤하게 선택하여 B에게 암호화된 메시지를 보내기 위한 공개키로 이것을 이용한다. DEA가 그들 두 참여자간의 통신을 지속적으로 도청하였을 경우에 DEA는 확률 P 만큼 메시지를 얻게 된다.

여기에서 각 키위탁 기관은 2.3장에 기술된 키생성 프로토콜을 통해 공개키들 $V_i \equiv g^x$에 상응되는 개인키의 공유값을 지니게 된다. DEA는 멱승보간법(Exponentiation Interpolation Method)을 통해 $V_i \equiv g^x$을 계산할 수 있게 되고 $V_j \equiv V_i Y_u^{j-i}$ $(i \neq j)$을 설정할 수 있게 된다. 또한 누구든지 $V_j \equiv V_0 Y_u^j$ $(j \in Z_n)$을 검사하는 것이 가능하다.

3.2 프로토콜의 기본 시나리오

프로토콜의 기본 수행은 확률 $P = a/2^n$을 갖는 시스템으로 다수의 분배된 법집행기관으로 이루어진다. 여기에서 확률 P는 임의의 유한 이진 분수 값 $P = a/2^n = \sum_{i=1}^{n} a_i 2^{-i}$이다.

3.2.1 초기 시스템 설정 및 DEA의 공개키 공개 단계

① (사전 계산) DEA는 $a_1, \cdots, a_n \in \{0,1\}$을 선택하여 확률 P을 결정한 후 $a_1, \cdots, a_n$을 공개한다. 단 p, q, g, Y_u은 사전에 알려진다.

② DEA는 $b_1, \cdots, b_n \in_R \{0,1\}$을 선택하고 $x_1, \cdots, x_n \in_R Z_q$을 계산한다. 여기에서 모든 비밀값 x_i $(i = 1, \cdots, n)$은 각각 검증가능한 비밀공유기법을 이용한 키생성 기법을 통해 생성되며, DEA의 공개키 g^{x_i}는 각각 멱승보간법을 이용해 계산된다.

③ DEA는 V_i 또는 $\overline{V}_i$ 둘 중의 하나가 "정확한 키"이도록 계산한 후, 다음과 같은 키들의 i개 쌍들을 이루도록 한다.

$$(V_i, \overline{V}_i) = \begin{cases} \text{만일 } b_i = 0, & (g^{x_i}, Y_u g^{x_i}) \\ \text{만일 } b_i = 1, & (g^{x_i} / Y_u, g^{x_i}) \end{cases}$$

④ 검증가능한 키위탁 기법 [4,5]에 사용된 기술처럼, 각 DEA는 x_{DEA_i}을 생성하고 이에 대한 공개키 $y_{DEA} \equiv g^{x_{DEA}}$을 따로 정리한다. 이후 비밀키 x_{DEA_i}는 검증가능한 (t,n)-threshold 비밀공유기법을 통해 정확히 다시 생성 될 수 있다.

⑤ DEA는 공개키들 $(V_i, \overline{V}_i)$의 n개의 쌍을 순차적으로 묶어 공개하고, A에게 $y_{DEA} \equiv g^{x_{DEA}}$을 공개한다.

3.2.2 A와 B간의 통신 단계

① (사전 계산) A는 $a_1, \cdots, a_n \in \{0,1\}$과 확률 P을 알고 있다.

② 우선 A는 n개의 키 $K_1, \cdots, K_n \in_R Z_q$을 순차적으로 선택한 다음 그 값들 합을 계산한다. 즉 $L_0 = 0$, $L_i \equiv \sum_{i=1}^{n} K_i \pmod q$ $(i = 1, \cdots, n)$

③ 통신 단계에서, A는 B에게 암호화된 메시

지를 전송할 것이다. 이를 위해 A는 자신의 세션키 $S \in_R Z_q$을 이용한다. 따라서 A는 대칭키로 암호화된 메시지 $E_S(M)$과 B의 공개키 y_B로 암호화된 세션키 $E(S, y_B)$을 준비한다.

④ A는 n개의 랜덤한 비트열 이루어진 $r_1, \cdots, r_n \in_R \{0,1\}$과 $e_1, \cdots, e_n \in_R Z_{q-1}$을 선택한다.

⑤ A는 다음 식과 같이 J_i를 설정한다.

$$J_i = \begin{cases} \text{만일 } a_i = 0, & 0 \\ \text{만일 } a_i = 1, & S + L_{i-1} \end{cases}$$

설정 후 $(c_1^{(i_j)}, c_2^{(i_j)})$와 $(c_1^{(i_k)}, c_2^{(i_k)})$을 계산한다.

만일 $r_i = 0$,

$$\begin{cases} E(J_i, V_i) = (c_1^{(i_j)}, c_2^{(i_j)}) = (g^{e_i}, J_i V_i^{e_i}) \\ E(K_i, \overline{V}_i) = (c_1^{(i_k)}, c_2^{(i_k)}) = (g^{e_i}, K_i \overline{V}_i^{e_i}) \end{cases}$$

만일 $r_i = 1$,

$$\begin{cases} E(J_i, \overline{V}_i) = (c_1^{(i_j)}, c_2^{(i_j)}) = (g^{e_i}, J_i \overline{V}_i^{e_i}) \\ E(K_i, V_i) = (c_1^{(i_k)}, c_2^{(i_k)}) = (g^{e_i}, K_i \overline{V}_i^{e_i}) \end{cases}$$

⑥ 최종적으로 $r_i, (c_1^{(i_j)}, c_2^{(i_j)}), (c_1^{(i_k)}, c_2^{(i_k)}), E_S(M), E(S, y_B)$을 A가 B에게 전달한다.

3.2.3 DEA에 의한 복호화 단계

만일 DEA가 V_i의 이산대수 x_i를 알고 있다면, ElGamal 암호 알고리즘으로 암호화된 세션키 S를 2.3장에서 논의된 기법을 통해 계산해 낼 수 있게 된다.

3.3 기본 시나리오의 확장

[7]에 공개적으로 검증가능한 부분 키위탁 기법이 제안되었다. 이 부분 키위탁 기법에서는 A가 공개키를 생성하여 공개하고, 개인키의 일부를 위탁한 후 그 나머지 부분은 위탁하지 않도록 구성된다. 동시에 A는 위탁하지 않는 부분(비트열)에 대한 증명을 수행하여야 한다. 개인키의 위탁되지 않은 부분은 현재의 컴퓨터 연산 속도 및 [7]에 논의된 사항에 따라 대략 $l = 80$ 비트 정도로 한정된다.

A가 위탁된 키 설정 요청시에, A는 단계 (1)에서 $K_i = K_i^{(0)} + K_i^{(1)}$ $(i = 1, \cdots, n)$을 만족하는 개인키 $K_1, \cdots, K_n \in_R Z_q$을 선택한다. 여기에서 비위탁된 부분의 비트 길이는 $|K_i^{(0)}| = l$ 이 된다. A의 공개키는 $Y_i = f^{K_i} = f^{K_i^{(0)}} f^{K_i^{(1)}} = Y_i^{(0)} Y_i^{(1)}$가 되며, $K_i^{(0)}$와 $K_i^{(1)}$는 다음과 같은 이진표현 형태로 표현할 수 있다.

$$\begin{cases} K_i^{(0)} = k_{(i,0)}^{(0)} 2^0 + \cdots + k_{(i,l-1)}^{(0)} 2^{l-1} \\ \text{여기에서, } k_{(i,j)}^{(0)} \in \{0,1\},\ j = 0, \cdots, l-1 \end{cases}$$

$$\begin{cases} K_i^{(1)} = k_{(i,0)}^{(1)} 2^0 + \cdots + k_{(i,m)}^{(1)} 2^m \\ \text{여기에서, } k_{(i,j)}^{(1)} \in \{0,1\},\ j = 0, \cdots, m \end{cases}$$

① $|K_i^{(0)}| = l$ 임을 증명하기 위해, A는 ElGamal 공개키 암호로 암호화된 $E(f^{k_{(i,j)}^{(0)}}, y_{DEA})$을 다음과 같이 구성한다. 여기에서 $w_{(i,j)}^{(0)} \in_R Z_q$이다.

$$E(f^{k_{(i,j)}^{(0)}}, y_{DEA}) = (A_{(i,j)}^{(0)}, B_{(i,j)}^{(0)}) = (g^{w_{(i,j)}^{(0)}}, y_{DEA}^{w_{(i,j)}^{(0)}} f^{k_{(i,j)}^{(0)}})$$

또한 다음과 같이 $A_i^{(0)}$와 $B_i^{(0)}$을 정의한다. 단, $W_i^{(0)} = \sum_{j=0}^{l-1} w_{(i,j)}^{(0)} 2^j$ 이고 $Y_i^{(0)} = f^{(\sum_{j=0}^{l-1} k_{(i,j)}^{(0)} 2^j)}$이다.

$$A_i^{(0)} = \prod_{j=0}^{l-1} (A_{(i,j)}^{(0)})^{2^j} = \prod_{j=0}^{l-1} (g^{w_{(i,j)}^{(0)}})^{2^j} = g^{(\sum_{j=0}^{l-1} w_{(i,j)}^{(0)} 2^j)} = g^{W_i^{(0)}}$$

$$B_i^{(0)} = \prod_{j=0}^{l-1} (B_{(i,j)}^{(0)})^{2^j} = \prod_{j=0}^{l-1} (y_{DEA}^{w_{(i,j)}^{(0)}} f^{k_{(i,j)}^{(0)}})^{2^j} = y_{DEA}^{(\sum_{j=0}^{l-1} w_{(i,j)}^{(0)} 2^j)} f^{(\sum_{j=0}^{l-1} k_{(i,j)}^{(0)} 2^j)} = y_{DEA}^{W_i^{(0)}} Y_i^{(0)}$$

② $E(Y_i^{(1)}, y_{DEA}) = (A_i^{(1)}, B_i^{(1)}) = (g^{W_i^{(1)}}, y_{DEA}^{W_i^{(1)}} Y_i^{(1)})$을 A가 B에게 전송한다. 이때 DEA는 $K_i^{(0)}$의 1 비트마다 그림 3에 도시된 비대화형 프로토콜을 수행하게 된다.

③ 위의 유사하게 A는 시민연합의 공개키 Y_u로 부분 개인키 $K_i^{(1)}$을 암호화 할 수 있다. 결국 A는 $E(Y_i^{(1)}, y_{DEA})$을 계산한 후, Y_u로 암호화된 메시지 $E((A_i^{(1)}, B_i^{(1)}) \| W_i^{(0)} + W_i^{(1)}, Y_u)$을 B에게 보내게 된다. $\|$는 비트 연결을 의미한다.

3.4 세션키 복구 및 메시지복구 단계

두 통신 참여자가 범죄행위를 행하고 있음에 대한 명백한 증거가 제시되는 경우, 이에 대한 대비책이 필요하다. 이를 본 논문에서는 세션키 복구 및 메시지 복구 단계라 정의하며, 이 단계에서는 DEA가 믿을 수 있는 제3자의 비정부기관(또는 시민단체)와 연합하여 두 통신 참여자가 사용하였던 세션키 복호화하는 것에 그 목적이 있다.

① A는 B에게 $r_i, (c_1^{(i_j)}, c_2^{(i_j)}), (c_1^{(i_k)}, c_2^{(i_k)}), (A_i^{(0)}, B_i^{(0)}), E((A_i^{(1)}, B_i^{(1)}) \| W_i^{(0)} + W_i^{(1)}, Y_u), E_S(M), E(S, y_B)$을 전달한다. 여기에서 DEA는 각 세 개의 파라메터 $r_i, (c_1^{(i_j)}, c_2^{(i_j)}), (c_1^{(i_k)}, c_2^{(i_k)})$의 암호문을 복호화 하는 것이 가능하다. 즉 DEA는 i 번째에서 $a_i = 0$ 이면 0 또는 K_i을 얻게 되고, $a_1 = 1$이라면 DEA는 $S + L_{i-1}$ 또는 K_i을 얻게 된다. 이것은 DEA가 a_i와 r_i을 알고 있

기 때문에 0 또는 $S+L_{i-1}$ 둘 중에 하나만을 갖게 된다. 만일 DEA가 $S+L_{i-1}$을 통해 $t-1$ $(1\le t\le n)$개의 키를 갖는다면 DEA는 S를 얻게 된다. 여기에서 $a_i=1$ 이 t번 발생하면 확률은 2^{-t}가 됨에 따라 DEA는 확률 $P=\sum_{i=1}^{n} a_i 2^{-i}$로 S을 얻게 된다.

② B는 메시지 M과 세션키 S을 얻는다. 만일 B이 자신의 비밀키로 $E(S, y_B)$을 복호화 할 수 있다면, B는 메시지 M을 얻을 수 있게 된다.

③ 위에 기술된 방법은 두 참여자간의 범죄적 행위에 대한 명확한 증거가 제시되는 경우라도 확률 $1-P$ 범위의 세션키는 기본적으로 키복구가 불가능하다. 따라서 이 같은 경우에는 세션키는 일부분이라도 노출시킬 필요가 있다. 일단 노출이 되면 이에 대한 다양한 공격(예로 전사적 공격)을 수행함으로써 세션키를 복구하는 것이다.

이를 위해 DEA는 믿을 수 있는 제3의 기관에게 $E((A_i^{(1)}, B_i^{(1)}) \| W_i^{(0)} + W_i^{(1)}, Y_u)$의 복호를 요청하고, $(A_i^{(1)}, B_i^{(1)})$와 $W_i^{(0)} + W_i^{(1)}$을 얻게 된다. DEA는 $W_i^{(0)} + W_i^{(1)}$을 알게 됨에 따라 $A_i^{(0)} A_i^{(1)} = g^{(W_i^{(0)} + W_i^{(1)})}$을 검증할 수 있고 또한 $Y_i = y_{DEA}^{-(W_i^{(0)} + W_i^{(1)})} B_i^{(0)} B_i^{(1)}$을 구할 수 있게 된다. DEA는 키복구에 사용되는 Y_i을 증명하고 $(A_i^{(0)}, B_i^{(0)})$와 $(A_{(i,j)}^{(1)}, B_{(i,j)}^{(1)})$ $(j=0, \cdots, m)$을 보관한다. 비밀정보 공유자들이 이 데이터들을 복호화하기 전까지, 부분 공개키 $Y_i^{(0)}$와 $Y_i^{(1)}$는 알 수 없으며 이는 정확한 키복구가 불가능하게 됨을 의미한다. 그러나 공유된 데이터의 복호화가 이루어지게 된다면 DEA는 $K_1, \cdots, K_n$을 얻게 되고 이에 따라 S을 복구 할 수 있게 된다.

4. 결론

본 논문에서는 두 참여자의 암호화된 채널을 정해진 범위 내에서 확률적으로 깰 수 있고 더불어 두 참여자가 심각 불법행위를 하고 있다고 판단되는 경우 어떠한 정보라도 부분적으로는 노출되도록 구성된다. 즉 법집행기관이 두 통신 참여자가 불법적 행위를 하고 있다고 판단되는 경우 분배된 키위탁 기법에 따라 확률 P 만큼 세션키 S을 구할 수 있는 것이 가능하고 더불어 만일 상당히 큰 불법적 행위가 이루어진다고 판단되면 어떠한 세션키라도 세션키의 일부분은 반드시 노출되도록 구성되어 있다. 이점은 보안의 다양성 측면에 있어서 기존에 제안된 방법 [2] 보다 좋은 특성을 보여 주는 것이다. 그러나 ElGamal의 공개키 암호, 공개적으로 검증가능한 ElGamal의 공개키 암호 시스템의 공통지수 지식 증명, 분배된 키위탁 기술, 그리고 부분 키위탁 기법 등 다양한 암호 기술 적용으로 인해 데이터 계산량 복잡도 및 전체 시스템의 통신량이 증가되었다. 따라서 이에 대한 해결이 앞으로의 과제로 남겨져 있다.

참고문헌

[1] D. Chaum and T. Pedersem. Wallet databases with observers. In Advances in Cryptology Crypto92, Vol. 740, LNCS, Springer-Verlag, pp. 89-105, 1993.

[2] M. Bellare and R. Rivest, Translucent cryptography - An alternative to key escrow, and its implementation via fractional oblivious transfer. Earlier version was MIT Laboratory for Computer Science Technical Memo No. 683, February 1996.

[3] R. Cramer, R. Gennare and B. Schoenmakers. A secure and optimally efficient multi-authority election scheme. In Advances in Crytology Proceedings of EUROCRYPT97 (LNCS 1233), pp 103-118. Springer-Verlag, 1997.

[4] JongHo Ryu, YoumHeung Youl, Efficient Partial Information-Revealing Protocol for Key Escrow Scheme with Distributed Key Escrow Authorities, WISA2001, September, 2001.

[5] W. Mao, Publicly Verifiable Partial Key Escrow, Information and Communications Security, ICICS '97, Springer-Verlag, pp. 409-413.

[6] D. Chaum and T. Pedersen, Wallet databases with observer, In Advances in Cryptology Crypto92, Vol. 740, LNCS, Springer-Verlag, pp. 89-105, 1993.

[7] M. Stadler, J.-M. Piveteau, and J. Camenisch, Fair Blind Signatures, Advances in Cryptology Eurocrypto'95, Lecture Notes in Computer Science v. 921, pp. 209-219, Springer Verlag, 1995.