논문 02-27-12C-9 한국통신학회논문지 '02-12 Vol.27 No.12C

Translucent Cryptography의 취약점 개선

김 종 희*, 정회원 이 필 중**

The Improvement of Translucent Cryptography

Chong Hee Kim*, Pil Joong Lee** *Regular Members*

요 약

키 복구 시스템의 대안으로 Bellare와 Rivest가 translucent cryptography의 개념을 소개하였다. Translucent cryptography란 사용자간의 암호 통신을 제3자가 확률 p($0 \le p \le 1$)로 복구할 수 있는 시스템이다. 이 시스템에서는 사용자의 비공개키를 사전에 위탁할 필요가 없으며, p의 값을 조정함으로써 사용자와 복구 기관 사이의 균형을 조정할 수 있다. 예를 들어, p의 값을 0.4로 한다면 복구 기관은 사용자간의 암호 메시지 100개 중 40개를 복구할 수 있게 된다. 본 논문에서는 translucent cryptography에서 사용자가 손쉽게 복구 기관을 속여 복구 기관이 p의 확률만큼 복호화할 수 없는 문제점을 지적하고, 이에 대한 개선책을 제안한다.

ABSTRACT

Bellare and Rivest proposed the translucent cryptography which was a kind of key recovery system. Translucent cryptography is a system in which the third party can recover encrypted message with the probability p($0 \le p \le 1$). The key recovery agency doesn't need to store the user's private key in advance. The balance between key recovery agency and user can be controlled by the value of p. For example, if p is set to be 0.4 then the key recovery agency can recover 40 out of 100 encrypted messages. In this paper, we show that user can easily deceive the key recovery agency in the translucent cryptography. Then we propose the solution of this problem.

I. 서 론

정보의 디지털화가 급속히 증가하였고, 기업 내뿐 아니라 계열사 간 거래도 전자적으로 이루어지는 현황에서 정보 보호의 요구가 증대됨에 따라 암호화 기술이 많이 채택되고 있다. 암호화 기술은 안전한 비밀 통신 등을 보장하여 주지만, 한편으로는 범죄자에 의한 비밀 정보의 불법적 활용이나 사용자가 키를 분실하였을 경우와 같은 문제점을 야기시킨다. 이러한 문제점을 해결하기 위해 키 복구 시스템이 최근 몇 년간 활발히 연구되어 왔다[2][6][7][8][10][11]. 키 복구 시스템은 분실되거나 훼손된 키를 복구할 수 있는 서비스를 제공할 뿐만 아니라, 이를 확장하여 비밀 정보의 유출 등을 대비하여 정당한 제 3의 기관이 감사 시스템을 구축할 수 있게 해 준다.

만일 정부 등과 같은 공공기관이 키 복구 시스템을 악용할 경우, 개인이나 기업 간의 모든 비밀 통신을 감청할 수 있기 때문에, 사용자들은 가능한 자신의 비밀 통신이 키 복구 기관에 의해 복구되기를 원하지 않을 것이다. 반면 키 복구 기관에서는 모든 사용자의 비밀 통신 내용을 복호하기를 원할 것이다.

Bellare와 Rivest는 이러한 사용자와 키 복구기관 사이의 이해 관계를 만족시키기 위해 non-interactive oblivious transfer를 이용하여 새로운 키 복구

* 포항공과대학교 전자전기공학과(chhkim@oberon.postech.ac.kr), ** 포항공과대학교 전자전기공학과(pjl@postech.ac.kr)
※ 본 연구는 전자·컴퓨터공학부를 통한 교육부 두뇌한국21사업과 MSRC, Com2MaC-KOSEF의 지원에 의하여 수행되었습니다
※ 본 논문은 2002년 4월 JCCI 학술대회에서 우수논문으로 선정되어 게재 추천된 논문입니다.

시스템인 translucent cryptography를 제안하였다[9]. 이 시스템에서는 키 복구 기관이 확률 $p(0 \le p \le 1)$로 사용자간의 비밀 통신을 복구 할 수 있다. 즉, p=0.4라면, 키 복구 기관은 사용자간의 100개의 암호 메시지 중 40개의 메시지를 복구할 수 있게 된다.

본 논문에서는 translucent cryptography에서 사용자가 손쉽게 키 복구기관이 비밀 메시지를 복구할 수 없도록 하는 문제점에 대해 언급하고, 이 문제점의 개선책을 제안한다. 본 논문의 구성은 다음과 같다. 먼저 2장에서는 non-interactive oblivious transfer에 대해 살펴보고, 이를 이용한 translucent cryptography를 3장에서 설명한다. 4장에서는 어떻게 사용자가 손쉽게 키 복구 기관을 속일 수 있는가에 대해 설명하고, 이의 개선책을 제안한 후, 5장에서는 결론을 맺는다.

Ⅱ. Non-interactive Oblivious Transfer

Alice와 Larry라는 두 명의 사용자가 다음과 같은 조건을 만족하며 통신을 한다면 이러한 프로토콜을 Oblivious transfer(OT)라고 한다[1].

- Larry는 확률 1/2로 Alice로부터 메시지를 받을 수 있다.
- Alice는 Larry가 메시지를 받았는지 여부를 알 수 없다.

Alice와 Larry간의 통신이 상호작용에 의해 발생하지 않고 Alice로부터 일방적으로 Larry에게 메시지를 보낸다면, 이러한 프로토콜을 Non-interactive oblivious transfer(NIOT)라고 한다.
Bellare와 Micali는 다음과 같은 NIOT프로토콜을 제안하였다[4].

- 초기화 단계

소수 ρ에 대해, Z^*_ρ의 생성자를 g라고 하자. U를 그 이산대수 값인 $\log_g(U) \bmod \rho$를 아무도 모르는 수라고 한다.

- 공개키 생성 단계

Larry는 b={0,1}의 값을 임의로 선택한 다음, g의 위수보다 작은 임의의 값 xb를 선택하여, $V_b = g^{xb} \bmod \rho$를 생성한 다음, 다음과 같이 (V_1, V_2)쌍을 계산한다.

$$(V_1, V_2) = (g^{xb}, Ug^{xb}) \quad \text{if } b = 0$$
$$= (g^{xb}/U, g^{xb}) \quad \text{if } b = 1$$

Larry는 결국 $V_2 = V_1 U \bmod \rho$를 만족하는 (V_1, V_2)쌍을 생성한 것이 되며, 이 값을 공개한다. 이 때, Larry는 V_1의 이산대수 값 $\log_g(V_1) \bmod \rho$과 V_2의 이산대수 값 $\log_g(V_2) \bmod \rho$값 중 하나 만을 알게 된다.

- 통신 단계

Alice가 Larry에게 보내고자 하는 메시지를 s라고 하면, Alice는 Larry의 공개키 (V_1, V_2) 중 하나를 임의로 선택해서 다음과 같이 ElGamal암호화 방법[3]으로 암호화해서 Larry에게 보낸다.

$$E(s, V_i) = (C_1, C_2) = (g^y, sV_i^y)$$

공개키 생성 단계에서 Larry는 V_1과 V_2 중 하나의 이산대수 값만을 알게 되는데, 그 이유는 다음과 같다. $V_1 = g^{x1} \bmod \rho$, $V_2 = g^{x2} \bmod \rho$라고 하자. 만일 Larry가 2개의 이산대수 값인 x1, x2를 모두 알고 있다면 $V_2 = V_1 U \bmod \rho$의 관계로부터 $U = g^{(x2-x1)} \bmod \rho$가 되어, Larry는 U의 이산대수 값을 알 수 있게 된다. 이는 U의 이산대수 값은 아무도 모른다고 가정에 모순됨으로 Larry는 V_1과 V_2 중 하나의 이산대수 값만을 알게 된다.

Alice의 입장에서는 Larry가 올바르게 공개키 쌍을 생성했는가는 (V_1, V_2)쌍이 $V_2 = V_1 U \bmod \rho$관계식을 만족하는 지 확인하면 된다.

지금까지 살펴본 NIOT에서는 Larry가 메시지를 받을 확률이 1/2이 된다. 이 확률 p가 0에서 1사이의 임의의 값을 가질 때를 p-NFOT 즉, non-interactive factional oblivious transfer라고 한다.

Ⅲ. Translucent Cryptography

p-NFOT가 있다고 가정할 때, translucent cryptography를 어떻게 구성할 수 있는가에 대해 먼저 살펴 본 후, 어떻게 p-NFOT를 구성하는지에 대해 살펴보자.

3.1 Translucent Cryptography[9]

Alice와 Bob이라는 두 사용자가 비밀 통신을 한다고 가정하자. 그리고, Larry라는 제3자가 둘 사이의 비밀 통신을 감청하여 그 내용을 복호하고자 하는 대상이라고 하자.

Larry는 자신의 공개키의 집합 $(V_1, V_2, \cdots, V_m)$을 공개한다. Larry는 이 공개키 집합 중 임의의 a개에 대해서는 그 공개키에 해당하는 비공개키를 알고 있으나, 나머지 m-a개에 대해서는 비공개키를 알지 못한다.

Alice는 Larry의 공개키 중 임의의 하나를 선택해서 다음과 같이 토큰을 생성한다.

$$e_s(m) \parallel E_B(s) \parallel LEAF$$

여기서, $e(\cdot)$는 대칭키 암호 시스템을 $E(\cdot)$는 공개키 암호 시스템을 나타낸다. 보내고자 하는 메시지가 m인 경우, 임의의 세션키 s를 생성하여 m을 s로 암호화한 후, s를 상대방의 공개키인 B로 암호화한다. 그리고, 감청하는 제3자가 세션키 s를 풀어볼 수 있도록 LEAF(law enforcement access field)를 생성하여 덧붙이게 된다.

Alice가 Larry의 공개키 중 임의의 하나를 선택하여 LEAF를 $(i, E(s,V_i))$와 같이 생성한다면, Larry는 확률 p=a/m로 메시지를 복호화할 수 있게 되는 것이다.

즉, p-NFOT 프로토콜이 있다면, 이를 이용하여 LEAF를 생성하게 되면 translucent cryptography가 된다.

3.2 p-NFOT

Bellar와 Rivest는 확률 p가 0에서 1사이의 임의의 값을 가지는 두 종류의 p-NFOT(Binary scheme, polynomial scheme)를 제안하였다[9]. 본 논문에서는 두 방법 중 binary scheme에 대해서만 자세한 설명을 하고, polynomial scheme에 대해서는 그 원리만 설명한다. 자세한 내용은 [9]을 참조하기 바란다.

· Binary scheme

확률 $p = 0.a_1a_2a_n = a_1/2 + a_2/2^{-2} + \quad + a_n/2^{-n}$, $a_i=\{0,1\}$라고 하자.

<키 생성 단계>

Larry는 2장에서의 공개키 쌍 생성 방법과 동일하게 다음과 같은 n쌍의 공개키 쌍을 생성한다.

$$(V_1, V_1'), (V_2, V_2'), \cdots, (V_n, V_n')$$

$$(V_i, V_i') = (g^{xi}, Ug^{xi}) \quad \text{if } b_i = 0$$
$$= (g^{xi}/U, g^{xi}) \quad \text{if } b_i = 1$$

여기서, $b_i=\{0,1\}$에서 임의로 선택하며, xi는 g의 위수보다 작은 범위 내에서 임의로 선택한다. 이렇게 공개키 쌍을 생성하면, Larry는 n쌍의 공개키 중 각 쌍에 대해서 하나의 공개키에 대응되는 비공개키 만을 알고 있게 된다.

<통신 단계>

Alice가 Bob에게 보내는 토큰은 앞서 설명한 바와 같이 다음과 같다.

$$e_s(m) \parallel E_B(s) \parallel LEAF$$

Binary scheme의 경우, $LEAF = (T_1, T_2, \cdots, T_n)$이 된다. 각각의 T_i는 다음과 같이 생성된다.

Alice는 n개의 키인 $K_1, K_2, \cdots, K_n$을 선택한 다음, 다음과 같이 L_i, J_i를 계산한다.

$L_0 = 0$

$L_i = K_1 + K_2 + \cdots + K_i$ for i=1,2,,n

$J_i = 0$ if $a_i = 0$, "junk"

$s + L_{i-1}$ if $a_i = 1$, "jewel"

Alice는 n 개의 임의의 비트 $r_1, r_2, \cdots, r_n$을 선택하여, 다음과 같이 T_i를 생성한다.

만일 $r_i = 0$ 이면

$$T_i = (0, E(J_i,V_i), E(K_i, V_i'))$$

만일 $r_i = 1$ 이면

$$T_i = (1, E(J_i,V_i'), E(K_i, V_i))$$

<키복구 단계>

Larry는 각각의 T_i로부터 J_i 또는 K_i 중 하나의 값을 알 수 있다. 즉, 만일 a_i = 0이면 Larry는 junk(0) 값 또는 키(K_i)를 알 수 있게 되며, a_i = 1 이면 jewel(s + L_{i-1}) 값 또는 키(K_i)를 알 수 있게 된다.

따라서 $1 \le t \le n$인 t에 대해, Larry는 t-1개의 키 값을 받은 다음, jewel을 받게 되면 s를 복구할 수 있게 됨으로, Larry가 키를 복구할 확률은 $p = 0.a_1a_2a_n = a_1/2 + a_2/2^{-2} + \cdots + a_n/2^{-n}$가 된다.

· Polynomial scheme

Larry는 자신의 공개키의 집합 (V_1, V_2, … , V_m, W_0, W_1, … , W_a,)을 공개한다. 이 중 W_0, W_1, , W_a는 Larry가 올바르게 공개키들을 생성했는지 확인하기 위해 필요한 값들이다. Larry는 V_1, V_2, … , V_m 중 임의의 a개에 대해서는 그 공개키에 해당하는 비공개키를 알고 있으나, 나머지 m-a개에 대해서는 비공개키를 알지 못한다.

Alice는 Larry의 공개키 중 임의의 하나를 선택해서 다음과 같이 토큰을 생성한다.

$$e_s(m) \parallel E_B(s) \parallel LEAF$$

LEAF = (i, $E(s,V_i)$) 가 되며, 따라서 Larry는 확률 p=a/m로 메시지를 복호화할 수 있게 된다.

Ⅳ. Translucent Cryptography의 개선

3장에서 살펴 본 바와 같이, 확률 1/2인 NIOT를 바탕으로 p-NIOT를 구성하여 translucent cryptography를 만들었다.

사용자 Alice는 LEAF를 본인이 생성하게 됨으로, Larry가 공개한 공개키쌍 (V_1, V_2) 중 어느 하나의 공개키를 이용하여 LEAF를 만드는 것이 아니라, 제3의 공개키 값인 V_3를 이용하여 손쉽게 translucent cryptography 시스템을 무력화 시킬 수 있다.

물론 키 복구 시스템에서 double encryption[12]과 같은 방법을 사용하게 된다면, 모든 키 복구 시스템은 안전할 수 없으므로, 어느 정도 사용자가 프로토콜을 올바르게 사용한다는 가정이 필요하다. 하지만, translucent cryptography의 경우 Alice의 입장에서는 키 복구 시스템을 무력화시키기 너무 쉽다는 점이 문제가 된다.

따라서, 이러한 문제점을 Binding ElGamal 시스템[6]에서 사용한 방법을 응용하여 해결할 수 있다. 이해를 돕기 위해 먼저 translucent cryptography에서 확률 p가 1/2인 경우를 생각해 보자. 이 경우, Larry의 공개키 쌍은 (V_1, V_2)이 되며, Larry는 둘 중 하나의 공개키에 대한 비공개키를 알고 있다. Alice가 Bob에게 보내는 토큰은 다음과 같이 구성할 수 있다.

$$e_s(m) \parallel E_B(s) \parallel LEAF$$

$$LEAF = E_{Vi}(s) \parallel bind, \quad i = \{1,2\}$$

즉, 토큰을 다시 자세히 풀어서 기술하면

$$e_s(m) \parallel (g^k, sY_B^k) \parallel (sV_i^k) \parallel (g^j, (V_i/Y_B)^j, z=wk+j)$$

j와 w, k는 임의의 값이다.

$C = g^k$, $R = sY_B^k$, $R_v = sV_i^k$, $D = g^j$, $F = (V_i/Y_B)^j$ 로 두면, 다음과 같은 검증식을 이용해 Alice가 (V_1, V_2)가 아닌 임의의 V_3를 사용할 경우 Bob 또한 올바른 s값을 받을 수 없게 됨을 검증할 수 있다. 이 검증식은 g^k 와 $(V_i/Y_B)^k$ 를 만족하는 k를 Alice가 알고 있음을 증명하는 과정으로부터 유도된 것이다[5].

검증식:

$$g^z = C^wD$$

$$(V_i/Y_B)^Z = (R_v/R)^WF$$

즉, 만일 Alice가 V_3를 사용한다면, $R_v = sV_3^k$, $F = (V_3/Y_B)^j$가 되고, 결국

$$(V_i/Y_B)^Z \neq (R_v/R)^WF = (sV_3^k/\ sY_B^k)\ ^w(V_3/Y_B)^j$$

가 되어 검증에 실패하게 된다. 여기서, i = {1,2}

이다.

bind의 크기를 줄이기 위해 (D,F,z)대신 (w,z)만을 보내어도 가능하기[6] 때문에 부가적인 전송량은 g의 위수 크기의 2배가 된다.

확률 p가 1/2이 아닌 일반적인 경우인 binary scheme과 polynomial scheme의 경우를 살펴보자. polynomial scheme의 경우 $V_1, V_2, \cdots, V_m$ 중 임의의 1개의 공개키를 Alice가 선택하여 이 공개키로 s를 암호화하여 LEAF를 생성하기 때문에 이 경우는 p = 1/2인 경우가 동일한 방법으로 bind 정보를 생성하면 된다.

binary scheme의 경우, n번의 암호화가 필요로 하며 각각의 T_i들에서 s를 직접적으로 암호화하지 않고 J_i나 K_i들을 암호화하기 때문에 bind 정보를 붙이기 위해서는 polynomial scheme에 비해 많은 부가적인 전송량이 필요하게 된다.

V. 결 론

본 논문에서는 확률 $p(0 \le p \le 1)$로 키 복구 기관이 사용자간의 암호화된 통신을 복구할 수 있는 translucent cryptography의 취약점을 살펴보았다. 즉, 사용자가 고의적으로 키 복구 기관의 공개키가 아닌 임의의 키를 사용하여 LEAF를 생성할 경우, 이의 검증이 불가능하기 된다. 이러한 문제점을 개선하기 위하여 검증 정보를 부가적으로 LEAF에 포함시킨 개선책을 제시하였다.

참 고 문 헌

[1] M.Rabin, "How to exchange secrets by oblivious transfer," *Technical Report TR-81*, Harvard Aiken Computation Laboratory, 1981.

[2] S.T.Walker, S.B.Lipner, C.M.Ellision, D.K.Branstad, and D.M.Balenson, "Commercial Key Escrow: Something for Everyone Now and for the Future," *TIS Report 541*, Trusted Information Systems Inc., 1995

[3] T.ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inform. Theory*, 31, pp.469-472, 1985

[4] M.Bellare and S.Micali, "Non-interactive oblivious transfer and application," *CRYPTO* 89, pp.547-559, Springer-Verlag, 1990, LNCS 435.

[5] E.Verheul and H. van Tilborg, "Binding ElGamal: A Fraud-Detectable Alternative to Key-Escrow," *EUROCRYPT* '97, pp.119-133, 1997

[6] M.Bellare and S.Goldwasser, "Verifiable Partial Key Escrow," *The 4th ACM Conference on Computer and Communication Security*, , 1997

[7] M.Burmester, Y.G.Desmdt, and J.Seberry, "Equitable Key Escrow with Limited Time Span," *ASIACRYPT* '98, pp.380-391, 1998

[8] P.Paillier and M.Yung, "Self-Escrowed Public-Key Infrastructures," *ICISC* '99, pp.249-261, 1999

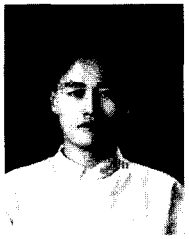
[9] M.Bellar and R.L.Rivest, "Translucent Cryptography? An Alternatvie to Key Escrow, and its Implementation via Fractional Oblivious Transfer," *Journal of Cryptology*, Vol.12, No2, pp.117-140, 1999.

[10] 유준석, 원동호, 이인수, 김병천, 박성준, "키 복구 시스템 및 안전성에 관한 고찰", *통신정보보호학회지* 제10권 1호, pp.21-37, 2000

[11] 유희종, 주미리, 원동호, 김지연, 박성준, "키 복구 시스템의 요구사항에 관한 고찰", *통신정보보호학회지* 제10권 1호, pp.1-19, 2000

[12] B.Pfitzmann and M.Waidner, "How to Break Fraud-Detectable Key Recovery", *Operating Systems Review*, 21, 1998, pp.23-28

김 종 희(Chong Hee Kim)



1997년 2월: 경북대학교 전자공학과 졸업
1999년 2월: 포항공과대학교 전자전기공학과 석사
1999년 3월~현재: 포항공과대학교 전자전기공학과 박사과정

<주관심 분야> 정보보안, 암호학

이 필 중(Pil Joong Lee) 정회원

1974년 2월 : 서울대학교 전자공학과 졸업
1977년 2월 : 서울대학교 전자공학과 석사 졸업
1982년 6월 : U.C.L.A. System Science, Engineer
1985년 6월 : U.C.L.A. Electrical Engineer, Ph.D.
1980년 6월~1985년 8월 : Jet Propulsion Laboratory, Senior Engineer
1985년 8월~1990년 2월 : Bell Communications Research, M.T.S
1990년 2월~현재 : 포항공과대학교 전자전기공학과 교수
<주관심 분야> 정보보안, 부호이론, 통신공학