Signcryption을 이용한 EC 기반의 안전한 인증된 키 교환 프로토콜 연구

김 락 현*, 이 성 용*, 염 흥 열*

순천향대학교 공과대학 정보보호학과*

Secure Authenticated Key Exchange Protocol based on EC using Signcryption Scheme

Rack-Hyun Kim*, Sung-Yong Lee*, Heung-Youl Youm*

Department of Information Security, Soonchunhyang University*

요 약

Signcryption은 공개키 암호와 디지털 서명을 결합한 하이브리드 공개키 프리미티브이다. Signcryption은 디지털 서명 수행 후 암호화 과정을 수행하는 프로토콜보다 연산량과 통신량에서 월등한 우월성을 보여준다. 그러나 Signcryption 기법을 이용한 상호 인증 및 키 교환 프로토콜은 적용시스템(예를 들어 모바일 네트워크)에 따라 많은 연산량과 통신량(상대적으로 모바일과 무선 네트워크에 대해)이 부담이 된다. 이에 이동망과 모선 네트워크에 적합한 적은 연산량과 통신량의 특성을 가진 상호 인증 및 키 교환 프로토콜을 제안한다. 본 논문에서는 두 참여자 사이에서 참여자의 상호 인증과 안전한 통신을 위해 Signcryption을 이용한 EC 기반의 안전한 인증된 키 교환 프로토콜을 제안한다. 그리고 제안 프로토콜의 보안성을 증명하고 효율성을 비교한다.

Ⅰ. 서 론

Signcryption 기법은 인증성을 제공하는 서명과정 후, 비밀성을 제공하는 암호화과정을 수행하는 하이브리드 기법으로 서명기법과 암호기법을 효율적으로 결합한 기법이다. Signcryption은 하나의 논리적인 단계에 인증과 비밀성 두 가지 모두를 지원하는 기법으로 적은 연산량과 통신량을 유지하기 때문에 많은 연구가 이루어지고 있다[1-3]. 그러나 유선 네트워크와 대용량 서버로 구성된 네트워크와는 달리 Signcryption을 이동망과 무선 네트워크에 적용할 경우, 연산량과 통신량이 부담이 된다. 이에 적은 연산량과 통신량을 갖는 이동망과 무선 네트워크에서 두 참여자를 상호 인증하고 안전한 통신을 위해 신선한 키를 교환할 수 있는 프로토콜을 제안한다.

본 논문에서 제안한 프로토콜은 이와 같은 조건을 만족하기 위해 EC(Elliptic Curve)를 기반으로 하였고, 상호 인증을 위해 Signcryption을 이용하였으며, 두 참여자의 비밀 정보를 이용하여 신선한 키를 교환한다. 본 논문에서 제안한 프로토콜은 일반적인 두 사용자간의 통신 모델, 응용 서버와 클라이언트 통신 모델, 그리고 이동단말기와 서버 모델 등 다양한 통신 구조에 적용이 가능하다.

본 논문의 구성은 다음과 같다. Ⅱ장에서 관련 연구로서 Zheng이 제안한 Signcryption[1]과 이에 EC를 적용한 프로토콜을 설명한다[4]. Ⅲ장에서는 Zheng의 Signcryption을 이용한 키 교환 프로토콜에 EC를 적용하고, Signcryption을 이용한 EC 기반의 안전한 인증된 키 교환 프로토콜을 Ⅳ장에서 제안하며, Ⅴ장에서는 효율성과 기능성을 분석한

* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음

다. 마지막으로 VI장에서는 결론을 기술한다.

II. Signcryption과 EC-Signcryption

2.1 Signcryption

Zheng은 서명과 대칭키 암호화를 Z_p로 정의되는 곱셈군에서 논리적인 한 단계로 수행하는 Signcryption을 1997에 제안하였다. 그리고 1998년에는 타원 곡선상의 덧셈군을 기반으로 하는 기법을 제안하였다[1].

표 1은 Zheng이 제안한 Signcryption에서 사용되는 파라미터를 정의한 것이고, 그림 1은 [1]에서 제안한 Signcryption과 Unsigncryption의 기본 프로토콜이다.

표 1 : Signcryption의 공개 및 비밀 파라미터

Alice 의 파라미터	공개 정보	Bob의 파라미터
x_a : 개인키 y_a : 공개키 $(y_a = g^{x_a} \bmod p)$	p : 큰 소수 q : 큰 소수 (factor of $p-1$) g : $0 < g < p$ 이고, 차수 $q \bmod p$ G,H: 일 방향 해쉬 E,D: 대칭키 암·복호 알고리즘	x_b : 개인키 y_b : 공개키 $(y_b = g^{x_b} \bmod p)$

Alice(Signcryption)	Bob(Unsigncryption)
① 랜덤선택 $x \in_R \{1, .., q-1\}$	① $w = (y_a \cdot g^r)^{s \cdot x_b} \bmod p$
② $w = y_b^r \bmod p$	② $k = G(w)$
③ $k = G(w)$	③ $m = D_k(c)$
④ $r = H(m, bind\_info, w)$	④ $if\ (r \stackrel{?}{=} H(m, bind\_info, w))$
⑤ $s = x/(r + x_a) \bmod q$	m 수락
⑥ $c = E_k(m)$	
⑦ (c, r, s) 전송	

그림 1 : Signcryption 기본 프로토콜

이 때, Zheng은 Alice의 Signcryption 과정에서 ⑤의 s와 w의 계산과정에 따라 3가지 방법을 제시하고 있다. 다음 표 2는 3가지 기법을 비교 설명한다.

표 2 : s와 w의 계산과정에 따른 3가지 기법

Alice(Signcryption)	Bob(Unsigncryption)
1) $s = x/(r + x_a) \bmod q$	1) $w = (y_a \cdot g^r)^{s \cdot x_b} \bmod p$
2) $s = x/(1 + x_a \cdot r) \bmod q$	2) $w = (g \cdot y_a^r)^{s \cdot x_b} \bmod p$
3) $s = (x - x_a \cdot r) \bmod q$	3) $w = (g^s \cdot y_a^r)^{x_b} \bmod p$

2.1 EC-Signcryption

본 절에서는 타원곡선상의 덧셈군을 기반으로 하는 EC-Signcryption을 설명한다.

표 3은 EC-Signcryption에서 사용되는 파라미터를 설명하고, 그림 2는 EC 기반의 Signcryption 프로토콜이다.

표 3 : EC-Signcryption의 공개 및 비밀 파라미터

공개 파라미터	
C	: $GF(p^n)$에서 타원곡선 $(p \geq 2^{150},\ m=1$ 또는 $p=2$ 그리고 $m \geq 150)$
q	: $\lvert p^m \rvert$을 만족하는 크기의 큰 소수
G	: 위수 q에 대한 점
$hash$	: 일방향 해쉬 함수
KH	: 일방향 Keyed 해쉬
(E,D)	: 대칭키 암·복호 알고리즘

Alice's Key	
v_a	: Alice의 개인키 $(\in [1, ..., q-1])$
P_a	: Alice의 공개키 $(P_a = v_a G)$

Bob's Key	
v_b	: Bob의 개인키 $(\in [1, ..., q-1])$
P_b	: Bob의 공개키 $(P_b = v_b G)$

Alice (A)		Bob(B)
1. Random v $v \in [1, ..., q-1]$ $(K_1, K_2) = hash(v \cdot p_b)$ 2. for m, generate (c, r, s) $c = E_{K_1}(m)$ $r = KH_{K_2}(m)$ i) SDSS1 $s = (v \cdot (r + v_a)) \bmod q$ ii) SDSS2 $s = (v \cdot (1 + r \cdot v_a)) \bmod q$	c, r, s →	1. $u \leftarrow r, s, g, p, p_a, v_b$ $u = s \cdot v_b \bmod q$ 2. K_1, K_2 계산 i) SDSS1 $(K_1, K_2) = hash(u \cdot p_a + u \cdot r \cdot G)$ ii) SDSS2 $(K_1, K_2) = hash(u \cdot G + u \cdot r \cdot p_a)$ 3. $m = D_{K_1}(c)$ $if\ (KH_{K_2}(m) \stackrel{?}{=} r)$ Accept m

그림 2 : EC-Signcryption 프로토콜

III. Signcryption을 이용한 키 분배 프로토콜

두 참여자간에 상호인증 과정을 수행하고 안전한 키를 교환하기 위한 프로토콜이 많이 연구되고 있다. Zheng은 1998년 Signcryption을 이용하여, 두 사용자간에 위조 불가능하고 신선한 세션키 교환이 가능한 프로토콜을 제안하였다.[3] 제안한 논문에서 Zheng은 키를 교환하기 위해 두 가지 프

로토콜을 제안하였다. 이 두 프로토콜은 세션키의 신선성을 위해 랜덤 수를 이용하는 방법과 타임스탬프를 이용하는 방법으로 구분된다.

다음 그림 3과 4는 [4]에서 Zheng과 Imai가 제안한 Signcryption을 이용한 키 교환 프로토콜에 EC를 적용한 EC-DKEUN(EC-Direct Key Exchange Using a Nonce)과 EC-DKEUTS (EC-Direct Key Exchange Using a Time-Stamp) 프로토콜이다.

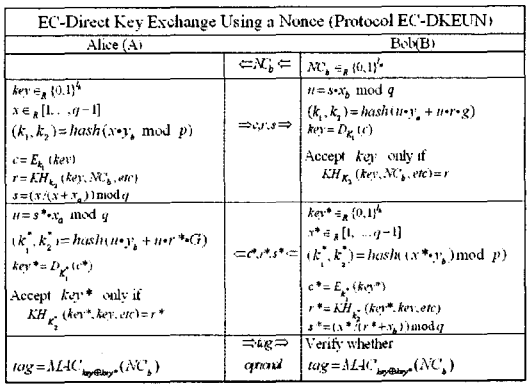

그림 3 : EC-DKEUN 프로토콜

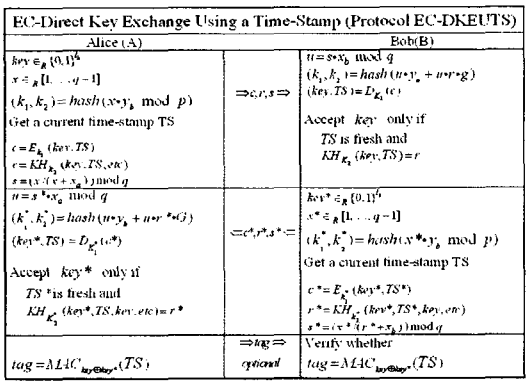

그림 4 : EC-DKEUTS 프로토콜

Ⅳ. SAKE와 EC-SAKE 프로토콜

본 논문에서는 Signcryption의 서명기법과 암호기법을 이용하여 참여자의 상호 인증 후, 안전한 키를 교환하는 프로토콜을 제안한다. 또한 연산량을 감소시켜 모바일과 무선 네트워크에서 적용 가능하도록 EC 기반의 Signcryption을 이용한 안전한 인증된 키 교환 프로토콜을 제안한다.

표 4는 제안 프로토콜에서 사용되는 파라미터를 정의한 것이고, 그림 5는 Signceyprion을 이용한 안전한 인증된 키 교환 프로토콜(SAKE : Secure Authenticated Key Exchange)이며, 그림 6은 Signcryption을 이용한 EC기반의 인증된 키 교환 프로토콜(EC-SAKE)이다.

표 4 : 제안 프로토콜에서 사용되는 공개 및 비밀 파라미터

Alice의 키	공개	Bob의 키
A:사용자 A의 ID x_a :개인키 y_a :공개키 $(y_a = g^{x_a} \bmod p)$ K_1, K_2 : 대칭키	p:큰 소수 q:큰 소수 (factor of $p-1$) g : $0<g<p$이고 차수 q mod p H_1, H_2, H_3 : 일방향 해쉬 KH : Key-ed 1-way 해쉬	B:사용자 B의 ID x_b :개인키 y_b :공개키 $(y_b = g^{x_b} \bmod p)$ K_1, K_2 : 대칭키

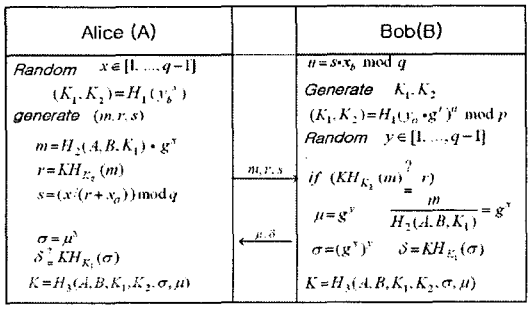

그림 5 : SAKE 프로토콜

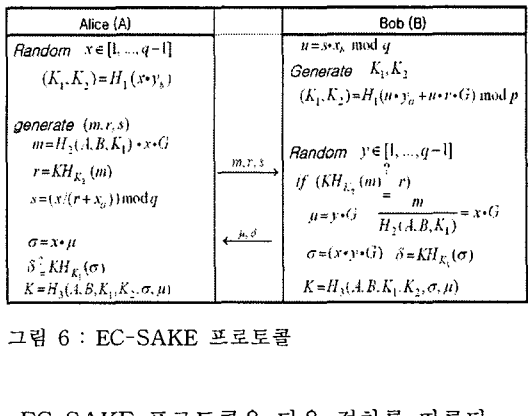

그림 6 : EC-SAKE 프로토콜

EC-SAKE 프로토콜은 다음 절차를 따른다.

1) Alice는 랜덤 정수 $x \in \{1, .., q-1\}$를 선택하고, 다음식에 의해 세션키 K_1, K_2를 계산

$$(K_1, K_2) = hash(x \cdot y_b)$$

2) 메시지 m에 대해, m, r, s을 생성

$$m = H(A, B, \pi) \cdot x \cdot G$$

$$r = KH_{K_2}(m)$$

$$s = (x/(r + x_a)) \bmod q$$

3) Bob에게 생성된 (m,r,s)를 전송

(m,r,s)를 수신한 Bob은 다음과 같이 Unsigncryption 절차를 수행한다.

1) 수신된 s를 이용하여 세션정보 u를 다음과 같이 계산

$$u = (s \cdot x_b) \bmod q$$

2) 세션 정보를 이용하여 세션키 K_1, K_2를 다음과 같이 계산

$$(K_1, K_2) = hash(u \cdot y_a + u r \cdot G) \bmod p$$

만일, $r \stackrel{?}{=} KH_{K_2}(m)$ 이 아니면, 세션이 종료되고, 조건에 만족하면 Bob은 Alice에게 인증 받기 위한 과정을 수행한다.

1) Bob은 자신의 비밀정보를 $\mu = y \cdot G$와 같이 계산하고, 수신된 m으로부터 Alice의 비밀정보를 계산하며, 다음 수식에 의하여 (μ, σ)를 계산한다. 이때 y는 랜덤 정수 $y \in \{1, .., q-1\}$이다

$$\mu = y \cdot G$$

$$\frac{m}{H(A, B, \pi)} = x \cdot G$$

$$\sigma = (x \cdot y \cdot G)$$

$$\delta = KH_{K_1}(\sigma)$$

2) Alice에게 (μ, σ)를 전송하고, Bob은 자신의 비밀 정보와 Alice의 비밀 정보를 이용하여 생성한 δ를 이용하여 세션 키를 생성

Alice는 Bob으로부터 수신한 (μ, σ)를 이용하여 자신의 비밀정보를 이용한 Bob의 메시지인 것을 확인 후, 세션키를 다음과 같이 생성

1) $\sigma = x \cdot \mu$

2) 다음 계산식과 같이 만일 수신된 δ와 Alice 자신이 계산한 값이 일치하면 Bob을 인증하고 세션키를 생성

$$\delta \stackrel{?}{=} KH_{K_1}(\sigma)$$

3) 마지막으로 Alice와 Bob의 세션 키는 다음 식에 의해 생성

$$K = H(A, B, K_1, K_2, \sigma, \mu)$$

V. 안전성 및 성능 분석

본 논문에서 제안한 프로토콜은 다음과 같은 보안 요구사항을 만족한다.

(1) 수동적 도청 공격(Passive Eavesdropping)

공격자는 두 참여자 사이에서 교환되는 메시지를 도청할 수 있고, 이들 사이에서 공유된 비밀정보와 통신 내용, 세션키를 구하려고 시도한다. 그러나 수동적 도청 공격자는 임의의 메시지를 바꾸거나, 삭제 또는 삽입하는 것이 불가능 하다. 이 공격은 EC-DLP의 어려움을 근거로 해결이 가능하다. 본 제안 프로토콜에서 공격자가 공개정보 p, q, g, y_a, y_b를 알고, Alice에서 Bob으로 전송되는 m, r, s를 알고 있다 하더라도, 공격자가 $x \cdot G$, $y \cdot G$를 구하는 것은 EC-DLP를 풀어내는 것만큼 어렵다. 또한 통신 주체간에 인증을 위한 파라미터와 해쉬값은 전송되지 않고 통신 주체가 직접 계산하므로 도청 공격을 통해서는 알 수 없다. 그리고 r, s로부터 x_a를 구하는 것은 Signcryption 안전성에 근거한다.

(2) 재전송공격(Replay attack)

재전송공격은 공격자가 Alice의 메시지 m을 Bob에게 재전송하여 이미 정상적인 Alice에 의해 생성된 이전 키(old session key)를 다시 생성하기 위한 공격이다. 그러나 모든 통신 메시지들은 매 세션마다 균일한 확률 분포에서 랜덤하게 생성된다고 가정하기 때문에 이 공격에 대한 공격자의 성공 확률은 무시할 만하다.

(3) 중간자 공격(Man-in-the-middle attack)

공격자가 두 참여자 사이에서 합법적으로 가장하거나 전송되는 메시지를 가로 챈 다음, 공격자와 Alice, 공격자와 Bob 사이에서 각각의 별도의 세션값을 만들어 내는 공격이다. 그러나 본 제안 프로토콜은 참여자의 비밀 정보를 유한 필드상에서의 EC-DLP 문제의 어려움에 근거하여 설계하였기 때문에, 공격자는 프로토콜 내의 모든 대화 내용을 이용하더라도 참여자의 비밀 정보를 알아낼 수 없다.

(4) 제안된 프로토콜은 오프라인(off-line) 사전 추측 공격에 대한 저항성을 갖는다.

제안된 프로토콜에서 공격자는 전송되는 메시지 m, r, s, μ 그리고 δ를 얻어도 DHP를 해결하지 못하므로 정확한 임의의 난수 x, y를 알 수 없다. 또한 전송되는 m, δ, K는 해쉬를 이용하여 인증되기 때문에 정확한 해쉬값을 모르면 이를 해결하지 못한다.

(5) PFS(Perfect Forward Secrecy)의 만족

PFS는 통신 주체들간에 장기간 개인키가 노출되더라도, 공격자가 통신 주체들간의 과거 세션키를 계산할 수 없는 경우, PFS를 만족한다고 정의한다. 제안 프로토콜은 각 세션마다 두 참여자의 비밀 정보와 난수를 이용하여 세션마다 세션키를 갱신하기 때문에 현재의 세션키와 이전의 세션키를 알 수 없다.

(6) 키의 신선성과 위조 불가능성

세션마다 사용자는 새로운 키 요소를 난수에 의하여 생성하고 이를 기반으로 세션키를 생성하므로 키의 신선성이 보장되고 이를 위조하는 것은 불가능하다. 그리고 참여자의 개인키 정보를 이용하여 상호 인증을 수행하고 이를 기반으로 세션키를 생성한다.

다음은 기존 프로토콜과 제안한 SAKE와 EC-SAKE 프로토콜의 계산적 비용과 기능면에서 성능을 비교 분석한 것이다.

표 6 : 기능적 비교

기법	상호 인증	통신량	키분배	Alice 키 요소	Bob 키 요소
DKEUN DKEUTS	○	3-way (+1opt)	○	○	×
EC-DKEUN EC-DKEUTS	○	3-way (+1opt)	○	○	×
SAKE EC-SAKE	○	2-way	○	○	○

표 7 : 연산량의 정량적 비교

기법 \ 연산량		계산적 비용							
		멱승	해쉬	암호	복호	곱셈	나눗셈	난수	TS
DKEUN	A	3	5	1	1	2	3	2	-
	B	3	5	1	1	2	3	1	-
DKEUTS	A	3	5	1	-	2	3	1	1
	B	3	5	-	1	2	3	1	1
SAKE	**A**	**3**	**5**	-	-	**1**	**1**	**1**	-
	B	**3**	**5**	-	-	**1**	**1**	**1**	-
EC-DKEUN	A	0	5	1	1	5	1	2	-
	B	0	5	1	1	5	1	3	-
EC-DKEUTS	A	0	5	1	1	5	1	2	1
	B	0	5	1	1	5	1	2	1
EC-SAKE	**A**	**0**	**5**	-	-	**4**	**1**	**1**	-
	B	**0**	**5**	-	-	**5**	**1**	**1**	-

표 6과 7에서 보는 바와 같이 Signcryption을 이용한 키 교환 프로토콜인 DKEUN, DKEUTS와 제안한 SAKE 프로토콜을 비교하면, 멱승과 해쉬의 연산량에서는 동일한 연산량을 보이나 암호와 복호에서 1~2번까지의 연산량 감소를 보인다. 또한 곱셈과 나눗셈에서도 1~2번의 연산량 감소를 보인다. 그리고 EC기반의 EC-DKEUN, EC-DKEUTS와 제안한 EC-SAKE 프로토콜을 비교하면 암호와 복호에서 1번의 연산량 감소를 보이고, 또한 곱셈과 나눗셈에서는 1~2번의 연산량 감소를 보이며, 난수 성성 연산에서 1번의 감소를 보여준다. 이와 같은 결과는 EC-SAKE를 모바일과 무선 네트워크에 상호 인증 및 키 교환프로토콜로서 적용하는 것이 가능함을 보여준다.

그리고 통신량 면에서 DKEUN, DKEUTS, EC-DKEUN 그리고 EC-DKEUTS 프로토콜은 상호 인증을 위해 3-way(+1:option) 통신량을 보이는 반면, 제안한 SAKE와 EC-SAKE 프로토콜은 2-way로써 상호 인증과 키 분배까지 가능하다.

Ⅵ. 결론

본 논문에서는 Signcryption을 이용하여 참여자 상호 인증 및 키 분배 프로토콜(SAKE)과 EC를 기반으로 한 안전한 키 교환 프로토콜(EC-SAKE)을 제안하였다. 이는 기존의 Signcryption 키 분배 프로토콜의 연산량과 통신량을 감소시키고, Signcryption을 이용하여 통신 당사자간의 인증과 참여자의 정보의 비밀성을 유지 하였다. 그 결과 모바일 네트워크와 무선 네트워크에서 참여자의 상호인증과 더불어 안전한 인증된 키를 교환하는 프로토콜로써 적용 가능함을 확인하였다.

향후, 사용자 인증이 가능하고 사용자 개인정보에 대한 비밀성을 유지하면서 사용자 정보를 기반으로 세션키를 세션마다 갱신하므로 도메인에서(하나 또는 다중 도메인에서) 안전하게 통신을 할 수 있는 "다중 도메인에서 상호 인증 및 안전한 키 교환 프로토콜"로 연구가 가능할 것으로 생각한다.

[참고문헌]

[1] Y. Zheng, "Digital signcryption or how to achive cost (signature and encryption) << cost(signature) + cost(encryption)", Advances in Cryptology, Proceedings of CRYPTO'97, LNCS Vol. 1294, Springer-Verlag, pp. 165-179, 1997

[2] Y. Zheng, "Updates on Signcryption" IEEE P1363, UCSB, 2002, 8

[3] Y. Zheng, "Shortened Digital Signature, Signcryption and Compact and Unforgeable Key Agreement Schemes" IEEE P1363a:Standard Specifications for Public-key Cryptography : Additional Techiques, 1998

[4] Y. Zheng, Hideki Imai, "How to construct efficient signcryption schemes on elliptic curves" Information Processing Letter 68, 227~233, 1998