Yang과 Chang의 기법의 수학적 결함과 그 해결책*

임원우, 오희국
한양대학교 컴퓨터공학과
e-mail: wonwoo@infosec.hanyang.ac.kr

Mathematical flaw of Yang and Chang's scheme and it's solution

Wonwoo Rhim, Heekuck Oh
Dept of Computer Science, Hanyang University

요 약

2009년 Yang과 Chang은 Computers and Security에 "An ID-based-remote mutual authentication with key agreement scheme on elliptic curve cryptosystem"을 제안하였다. 하지만 제안된 방법에서 사용한 타원곡선 곱셈에서 수학적 오류를 범하였고, 수학적 오류를 수정한 방법을 제안하고자 한다.

1. 서론

모바일 장비의 사용이 증가하고 있다. 언제 어디서든지 사용할 수 있다는 모바일 장비가 가지는 휴대성 때문이라고 할 수 있다. 하지만 그러한 특성 때문에 모바일 장비는 성능에 제약이 발생한다.

타원곡선 암호시스템을 이용한 다양한 인증 방법이 제안되고 있다. 공개키 암호시스템의 모듈러 지수 연산은 계산량이 크기 때문에 모바일 장비의 계산 능력이나 배터리 소모로 문제가 발생할 수 있다. 타원곡선 암호시스템은 보다 작은 크기의 키로 공개키 암호시스템과 같은 안전성을 가질 수 있기 때문에 계산량 측면에서 모바일 장비에 보다 적합하다. 예를 들어, 타원곡선 암호시스템의 160 bits 키는 공개키 암호시스템의 1024 bits의 키와 같은 안전성을 가진다[1].

2009년 Yang과 Chang은 Computers and Security에 신원 기반의 타원곡선 암호시스템을 이용한 "An ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem"을 제안하였다[2]. 하지만 YC 기법은 타원곡선에 정의되지 않는 연산을 사용하는 수학적 오류가 범하였다. 본 논문은 YC 기법에서 범한 수학적 오류를 수정하여 개선된 방법을 제안하였다.

* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2010-C1090-1011-0010).

* 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. 2010-0000438).

본 논문의 2장에서는 타원곡선 암호시스템에 대해서 살펴보고 YC 기법에 대해서 소개한다. 3장에서는 YC 기법의 수학적 오류를 수정한 방법을 제안하고 4장에서 수정한 과정에 대한 안전성을 분석하고 5장에서 결론을 맺는다.

2. 관련연구

이 장에서는 타원곡선 암호시스템에 대해서 알아보고 YC 기법에 대해서 소개한다.

2.1 타원곡선

타원곡선은 $y^2+axy+by=x^3+cx^2+dx+e$와 같은 3차원 방정식의 형태이다.(a, b, c, d, e는 실수이다.) 타원곡선 암호시스템에서 타원곡선 방정식은 소수 유한체 F_p상의 $E_p(a, b)$: $y^2=x^3+ax+b(\text{mod } p)$ 형식으로 정의된다.(a, b$\in F_p$, p>3, $4a^3+27b^2\neq 0(\text{mod } p)$) 정수 s$\in F_p^*$와 점 P$\in E_p(a, b)$가 주어졌을 때, $E_p(a, b)$상의 곱셈 s・P는 $s \cdot P=\underbrace{P+P+\cdots+P}_{s}$로 정의된다. 타원곡선 암호시스템의 보다 자세한 정의는 [1]에서 찾을 수 있다.

일반적으로 타원곡선 암호시스템의 안전성은 다음과 같은 문제의 어려움에 기반을 둔다[3].

정의 1. 타원곡선 이산대수 문제는 타원곡선 $E_p(a, b)$상의 두 점 P와 Q가 주어졌을 때, Q=s・P를 만족하는 정수 s$\in F_p^*$를 찾는 것이다.

정의 2. 타원곡선 Diffie-Hellman문제는 s, t$\in F_p^*$와 $E_p(a, b)$상의 세 점 P, s・P, t・P가 주어졌을 때, $E_p(a, b)$상의 점 (s・t)・P를 찾는 것이다.

정의 3. 타원곡선 인수분해 문제는 s, t$\in F_p^*$와 $E_p(a, b)$상의 두 점 P와 Q=s・P+t・P가 주어졌을 때, $E_p(a, b)$상의 두 점 s・P와 t・P를 찾는 것이다.

2.2 YC 기법

이 절에서는 YC 기법에 대해서 소개한다[2]. YC 기법은 시스템 초기화 단계, 사용자 등록 단계, 상호인증과 키 동의 단계의 3단계로 구성된다.

표 1은 YC 기법에서 사용된 표기법이다.

<표 1> YC 기법의 표기법

U	사용자
S	서버
ID_U	사용자의 신원정보
q_S, Q_S	서버의 개인키/공개키 쌍
A_{ID_U}	사용자의 인증키
Q_{ID_U}	사용자의 공개키
R_U	사용자가 선택한 타원곡선상의 점
R_S	서버가 선택한 타원곡선상의 점
k	세션키
T_i	현재 시간을 나타내는 timestamp
$H_i(\cdot)$	일방향 해시 함수

2.2.1 시스템 초기화 단계

단계 1. 서버는 타원곡선 방정식 $E_p(a, b)$와 위수 n을 선택한다.

단계 2. 서버 S는 타원곡선 상에서 위수가 n인 점 P를 선택한다(n은 안전을 위해 큰 수를 고려). 그 후, S는 $Q_S=q_S \cdot P$를 계산하여 자신의 공개키/개인키 쌍을 선택한다.

단계 3. 서버는 일방향 해시 함수 $H_1(\cdot)$: $\{0, 1\}\rightarrow G_P$, $H_2(\cdot)$: $\{0, 1\}\rightarrow Z_p^*$, $H_3(\cdot)$: $\{0, 1\}^*\rightarrow Z_p^*$를 선택한다($G_P$는 타원곡선 상에서 P에 의해 생성된 덧셈순환군).

단계 4. 서버는 q_S는 공개하지 않고 $\{E_p(a, b), P, Q_S, H_1(\cdot), H_2(\cdot), H_3(\cdot)\}$는 공개한다.

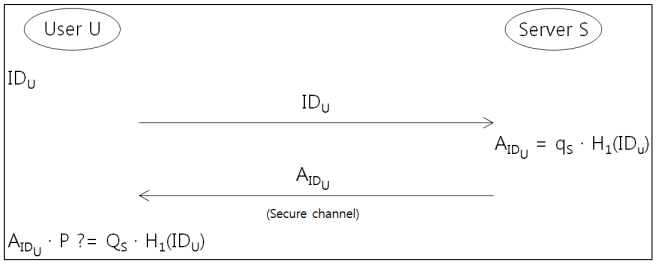
2.2.2 사용자 등록 단계

단계 1. 사용자 U는 자신의 신원정보 ID_U를 서버에 보낸다.

단계 2. 서버 S는 $A_{ID_U}=q_S \cdot H_1(ID_U) \in G_P$를 계산한다. 그 후, S는 안전한 채널을 사용하여 A_{ID_U}를 U에게 보낸다.

단계 3. A_{ID_U}를 받은 후, U는 $A_{ID_U} \cdot P=Q_S \cdot H_1(ID_U)$가 성립하는지 확인하고, 성립한다면 U는 A_{ID_U}를 비공개로 보관한다.

그림 1은 YC 기법의 사용자 등록 단계를 표현한 것이다.



(그림 1) YC 기법의 사용자 등록 단계

2.2.3 상호인증과 키 동의 단계

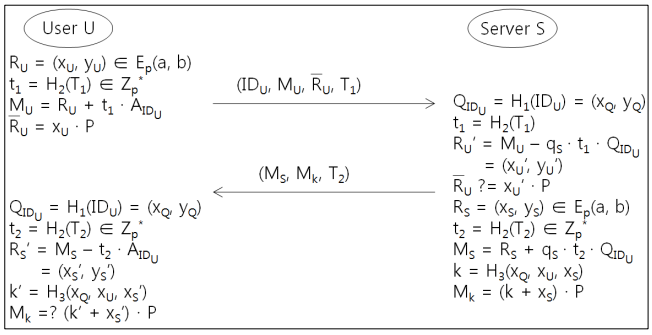
단계 1. 사용자 U는 임의의 점 $R_U=(x_U, y_U) \in E_p(a, b)$를 선택한다($x_U$와 y_U는 각각 점 R_U의 x와 y의 좌표). 그 후, U는 $t_1=H_2(T_1)$, $M_U=R_U+t_1 \cdot A_{ID_U}$, $\overline{R}_U=x_U \cdot P$를 계산한다. 최종적으로 U는 $(ID_U, M_U, \overline{R}_U, T_1)$을 서버에 보낸다.

단계 2. 서버 S는 $(ID_U, M_U, \overline{R}_U, T_1)$을 받은 후에 $Q_{ID_U}=(x_Q, y_Q)$와 $R'_U=(x'_U, y'_U)$를 얻기 위해 $Q_{ID_U}=H_1(ID_U)$, $t_1=H_2(T_1)$와 $R'_U=M_U-q_S \cdot t_1 \cdot Q_{ID_U}$를 계산한다. 그 후, S는 $\overline{R}_U=x'_U \cdot P$가 성립하는지 확인하고, 성립한다면 U가 정당한 사용자가 맞는다는 것과 $x'_U=x_U$ 임을 확인한다. 성립하지 않으면 프로토콜은 종료된다.

단계 3. 서버 S는 임의의 점 $R_S=(x_S, y_S) \in E_p(a, b)$를 선택한다. 그리고 $t_2=H_2(T_2)$, $M_S=R_S+t_2 \cdot q_S \cdot Q_{ID_U}$를 계산한다. 그 후, S는 $k=H_3(x_Q, x_U, x_S)$에 의해 세션키 k를 계산한다. 최종적으로 S는 $M_k=(k+x_S) \cdot P$를 계산하고 (M_S, M_k, T_2)를 U에게 보낸다.

단계 4. 사용자 U는 (M_S, M_k, T_2)를 받은 후에 $Q_{ID_U}=(x_Q, y_Q)$와 $R'_U=(x'_S, y'_S)$를 얻기 위해 $Q_{ID_U}=H_1(ID_U)$, $t_2=H_2(T_2)$와 $R'_S=M_S-t_2 \cdot A_{ID_U}$를 계산한다. 그 후, U는 $k'=H_3(x_Q, x_U, x'_S)$와 $M'_k=(k'+x'_S) \cdot P$를 계산하고 $M'_k=M_k$가 성립하는지 확인한다. 성립한다면 U는 서버가 맞는다는 것과 세션키 $k'=k$ 임을 확인한다. 성립하지 않으면 프로토콜은 종료된다.

그림 2는 YC 기법의 상호 인증과 키 동의 단계를 표현한 것이다.



(그림 2) YC 기법의 상호인증과 키 동의 단계

2.3 YC 기법의 문제점

2.1장의 내용과 같이 타원곡선에서의 곱셈은 덧셈의 반복으로 정의된다. 따라서 $Q=k \cdot P$에서 P는 (x, y)좌표로 표시할 수 있는 점을 의미하고 k는 P를 자기 자신에게 더하기 위한 횟수, 즉 상수가 된다. Q는 P와 같이 점이 된다.

YC 기법의 사용자 등록 단계에서 A_{ID_U}값의 검증을 위한 과정에 필요한 A_{ID_U}, P, Q_S, $H_1(ID_U)$ 값들은 모두 타원

곡선 상의 점을 나타낸다. 따라서 $A_{ID_U} \cdot P$와 $Q_S \cdot H_1(ID_U)$는 점·점으로 타원곡선에서 정의되지 않는 연산을 사용하였고 계산할 수 없는 오류를 범하였다.

3. 제안하는 방법

이 장에서는 기존의 방법인 YC 기법에 존재하는 타원곡선의 수학적 오류를 수정한 방법을 제안한다. 제안하는 방법은 시스템 초기화 단계, 사용자 등록 단계, 상호 인증과 키 동의 단계의 3단계로 구성되어 있다.

시스템 초기화 단계에서 서버의 공개키를 계산하는 과정에 서버의 개인키를 1개 추가하여 타원곡선 곱셈에서 타원곡선 덧셈과 곱셈을 혼합하는 방법으로 수정하였다. 사용자 등록 단계에서 서버에서 사용자의 인증키를 계산하는 과정도 타원곡선 곱셈에서 타원곡선 덧셈과 곱셈을 혼합하는 방법으로 수정하였다. 서버의 개인키 별로 각각 사용자의 인증키 값을 계산하여 사용자에게 전달하였고, 사용자는 전달받은 값들을 더하여 인증키 값을 검증하도록 하였다. 상호 인증과 키 동의 단계에서 서버의 개인키를 사용하는 부분의 타원곡선 곱셈을 타원곡선 곱셈과 덧셈을 혼합하는 방법으로 수정하였다.

표 2는 제안된 방법에 사용된 표기법이다.

<표 2> 표기법

U	사용자
S	서버
ID_U	사용자의 신원정보
$q1_S$, $q2_S$	서버의 개인키
Q_S	서버의 공개키
A_{ID_U}	사용자의 인증키
Q_{ID_U}	사용자의 공개키
R_U	사용자가 선택한 타원곡선상의 점
R_S	서버가 선택한 타원곡선상의 점
k	세션키
T_i	현재 시간을 나타내는 timestamp
$H_i(\cdot)$	일방향 해시 함수

3.1 시스템 초기화 단계

단계 1. 서버는 타원곡선 방정식 $E_p(a, b)$와 위수 n을 선택한다.

단계 2. 서버 S는 타원곡선 상에서 위수가 n인 점 P를 선택한다(n은 안전을 위해 큰 수를 고려). 그 후, S는 $Q_S=q1_SP+q2_SP$를 계산한다.

단계 3. 서버는 일방향 해시 함수 $H_1(\cdot)$: $\{0, 1\}\rightarrow G_P$, $H_2(\cdot)$: $\{0, 1\}\rightarrow Z_p^*$, $H_3(\cdot)$: $\{0, 1\}^*\rightarrow Z_p^*$를 선택한다($G_P$는 타원곡선 상에서 P에 의해 생성된 덧셈순환군).

단계 4. 서버는 $q1_S$과 $q2_S$는 공개하지 않고 $\{E_p(a, b), P, Q_S, H_1(\cdot), H_2(\cdot), H_3(\cdot)\}$는 공개한다.

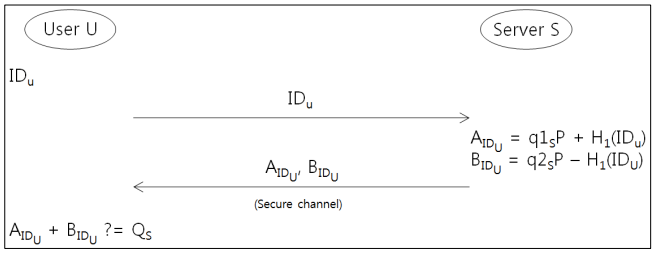
3.2 사용자 등록 단계

단계 1. 사용자 U는 자신의 신원정보 ID_U를 서버에 보낸다.

단계 2. 서버 S는 $A_{ID_U}=q1_SP+H_1(ID_U)\in G_P$, $B_{ID_U}=q2_SP-H_1(ID_U)\in G_P$를 계산한다. 그 후, S는 안전한 채널을 사용하여 A_{ID_U}와 B_{ID_U}를 U에게 보낸다.

단계 3. A_{ID_U}와 B_{ID_U}를 받은 후, U는 $A_{ID_U}+B_{ID_U}=Q_S$가 성립하는지 확인하고, 성립한다면 U는 A_{ID_U}와 B_{ID_U}를 비공개로 보관한다.

그림 3은 사용자 등록 단계를 표현한 것이다.



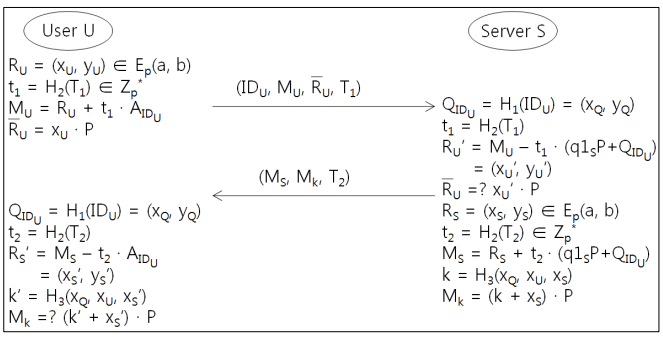
(그림 3) 사용자 등록 단계

3.3 상호 인증과 키 동의 단계

단계 1. 사용자 U는 임의의 점 $R_U=(x_U, y_U)\in E_p(a, b)$를 선택한다.($x_U$와 y_U는 각각 점 R_U의 x와 y의 좌표이다.) 그 후, U는 $t_1=H_2(T_1)$, $M_U=R_U+t_1 \cdot A_{ID_U}$, $\overline{R}_U=x_U \cdot P$를 계산한다. 최종적으로 U는 $(ID_U, M_U, \overline{R}_U, T_1)$을 서버에 보낸다.

단계 2. 서버 S는 $(ID_U, M_U, \overline{R}_U, T_1)$을 받은 후에 $Q_{ID_U}=(x_Q, y_Q)$와 $R'_U=(x'_U, y'_U)$를 얻기 위해 $Q_{ID_U}=H_1(ID_U)$, $t_1=H_2(T_1)$와 $R'_U=M_U-t_1 \cdot (q1_SP+Q_{ID_U})$를 계산한다. 그 후, S는 $\overline{R}_U=x'_U \cdot P$가 성립하는지 확인하고, 성립한다면 U가 정당한 사용자가 맞는다는 것과 $x'_U=x_U$ 임을 확인한다. 성립하지 않으면 프로토콜은 종료된다.

단계 3. 서버 S는 임의의 점 $R_S=(x_S, y_S)\in E_p(a, b)$를 선택한다. 그리고 $t_2=H_2(T_2)$, $M_S=R_S+t_2 \cdot (q1_SP+Q_{ID_U})$를 계산한다. 그 후, S는 $k=H_3(x_Q, x_U, x_S)$에 의해 세션키 k를 계산한다. 최종적으로 S는 $M_k=(k+x_S) \cdot P$를 계산하고 (M_S, M_k, T_2)를 U에게 보낸다.



(그림 4) 상호 인증과 키 동의 단계

단계 4. 사용자 U는 (M_S, M_k, T_2)를 받은 후에 $Q_{ID_U}=(x_Q, y_Q)$와 $R'_U=(x'_S, y'_S)$를 얻기 위해 $Q_{ID_U}=H_1(ID_U)$, $t_2=H_2(T_2)$와 $R'_S=M_S-t_2 \cdot A_{ID_U}$를 계산한다. 그 후, U는 $k'=H_3(x_Q, x_U, x'_S)$와 $M'_k=(k'+x'_S) \cdot P$를 계산하고 $M'_k=M_k$가 성립하는지 확인한다. 성립한다면 U는 서버가 맞는다는 것과 세션키 $k'=k$ 임을 확인한다. 성립하지 않으면 프로토콜은 종료된다.

그림 4는 상호 인증과 키 동의 단계를 표현한 것이다.

4. 안전성 분석

제안한 방법의 안전성은 타원곡선 이산대수 문제와 타원곡선 인수분해 문제의 어려움을 기반으로 한다. 임의의 공격자는 공개 값인 Q_S를 알 수 있지만 타원곡선 인수분해 문제의 어려움에 의해 $q1_SP$와 $q2_SP$를 계산할 수 없다. 만약 $q1_SP$와 $q2_SP$를 각각 알아낸다고 하더라고 타원곡선 이산대수 문제의 어려움에 의해 서버의 개인키는 $q1_S$와 $q2_S$를 계산할 수 없다.

사용자의 인증키를 계산하는 과정도 공개 값인 ID_U로 $H_1(ID_U)$를 계산할 수 있지만 $q1_S$와 $q2_S$를 알 수 없기 때문에 A_{ID_U}와 B_{ID_U}를 계산할 수 없다. 역으로 사용자의 인증키들의 합인Q_S를 알 수 있지만 서버의 개인키를 알 수 없는 것과 마찬가지로 A_{ID_U}와 B_{ID_U}를 계산할 수 없다.

5. 결론

YC 기법에 존재하는 수학적 오류를 수정한 방법을 제안하였다. YC 기법의 오류부분인 사용자 인증키 검증과정을 타원곡선 덧셈으로 구성하였다. 인증키 검증과정이 타원곡선 덧셈으로 이루어지도록 다른 과정을 타원곡선 곱셈에서 타원곡선 덧셈과 곱셈을 혼합하는 방법으로 구성하였다. 제안한 방법의 안전성 타원곡선 이산대수 문제와 타원곡선 인수분해 문제의 어려움을 기반으로 하여 또한 YC 기법과 동일한 안전성을 가진다.

참고문헌

[1] Hankerson D., Menezes A., Vanstone S., "Guide to elliptic curve cryptography," New York, USA: LNCS, Springer-Verlag, 2004.

[2] Yang J.H., Chang C.C., "An ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," Computers and Security, Vol. 28, pp. 138-143, 2009.

[3] Li F., Xin X., Hu Y., "Identity-based broadcast signcryption," Computer Standard and Interfaces, Vol. 30, pp. 89-94, 2008.

[4] Yoon E. J., Yoo K. Y., "Robust ID-based remote mutual authentication with key agreement scheme for mobile devices on ECC," International Conference on Computational Science and Engineering, Art no. 5283837, pp. 633-640, 2009.

[5] Prabu M., Shanmugalakshmi R., "A study of elliptic curve cryptography and its application," ICWET, pp. 425-427, 2010.