모바일 환경의 보안을 위한 ECC 기반의 빠른 키 생성 알고리즘 설계

윤성열*, 김현수**, 박석천***
*, **경원대학교 일반대학원 전자계산학과
***경원대학교 IT대학
e-mail:scpark@kyungwon.ac.kr

Design of ECC-based Fast Key Generation Algorithm for Security in Mobile Environments

Sung-Yeol Yun*, Hyun-Soo Kim**, Seok-Cheon Park***
*, **, ***Dept of Computer Science, Kyungwon University

요 약

모바일 환경에서 ECC 암호화 알고리즘을 사용할 때 키 교환을 위해 ECDH 알고리즘을 사용한다. ECDH 알고리즘에서 공개키를 생성할 때 난수와 타원곡선위의 점 G를 생성해야 한다. 이 때 타원곡선을 구성하는 파라미터 값의 크기가 클 경우, G를 생성할 때 좌표에 맞는 값을 구하는 시간이 오래 걸릴 수 있기 때문에 결과적으로 모바일 환경에서의 데이터 통신에 제약이 될 수 있다. 따라서 본 논문에서는 모바일 환경에서 ECDH 알고리즘의 키 생성 시간을 줄일 수 있는 알고리즘을 설계한다.

1. 서론

최근 스마트폰이 활성화됨에 따라 아이폰, 안드로이드폰, PDA 등의 모바일 단말에서 사용할 수 있는 어플리케이션 또한 기하급수적으로 증가하였다. 모바일 단말의 어플리케이션을 통해 전송되는 데이터는 주로 인터넷망을 이용하기 때문에 제3자에 의해 전송되는 데이터가 침해될 수 있다. 따라서 암호화를 통하여 데이터를 보호하는 것이 필요하다.

암호화 방식에는 공개키와 비밀키 알고리즘이 있는데, 데이터를 암호화하는 키와 복호화하는 키가 다른 공개키 알고리즘이 선호된다. 공개키 암호화 알고리즘에는 대표적으로 RSA와 ECC(Elliptic Curve Cryptography)암호화 알고리즘이 있는데, ECC 암호화 알고리즘의 160비트 길이의 키는 RSA의 1024비트 길이의 키와 보안의 강도가 같아서, 키 길이 대 보안효과가 더 효율적인 ECC 알고리즘이 선호된다. ECC 암호화 알고리즘은 ECDH(Elliptic Curve Diffie-Hellman)알고리즘을 이용하여 공개키를 교환한다. 공개키 생성 방법은 난수와 타원곡선위의 한 좌표 G를 이용하여 생성하게 되는데, 보안의 강도를 높이기 위해 파라미터 값들의 크기가 커질수록 공개키 생성 시간이 오래 걸리게 되어 결과적으로 제약적인 모바일 환경에서는 비효율적으로 운용될 수 있다. 따라서 본 논문에서는 모바일 환경에서도 ECC 알고리즘을 효율적으로 사용할 수 있도록 ECDH 알고리즘의 키 생성 시간을 줄이는 알고리즘을 설계한다.

2. ECDH

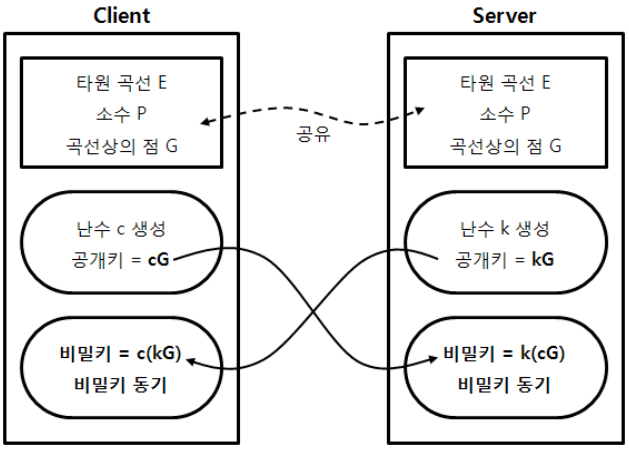
ECDH 알고리즘은 타원곡선암호(ECC) 알고리즘에서 사용하는 방법으로 데이터를 주고받으려는 2개의 단말이 비밀키를 생성하기 위해 사용된다. 비밀키 생성을 위한 공개키를 교환할 때 공통적으로 사용되는 파라미터는 표 1과 같다.

<표 1> ECDH에서 사용하는 파라미터 정의

파라미터	정의
p	타원곡선이 정의되는 유한체의 크기(소수)
E(GF(p))	유한체 GF(p) 위에서 a,b∈GF(p)에 의해 정의된 타원곡선
a, b	타원곡선을 결정하는 방정식의 상수
x, y	타원곡선 한 점의 좌표
c	클라이언트가 생성하는 난수
k	서버가 생성하는 난수
G	타원곡선 상의 임의의 점(좌표)

* 경원대학교 일반대학원 전자계산학과 박사과정
** 경원대학교 일반대학원 전자계산학과 석사과정
*** 경원대학교 IT대학 컴퓨터공학과 정교수(교신저자)

ECDH 알고리즘은 유한체 위의 Diffie-Hellman 알고리즘을 타원곡선 위에서 변환한 것으로 본질적으로는 Diffie-Hellman 알고리즘과 동작 방법이 같다. 그림 1은 난수와 곡선상의 임의 점 G를 이용하여 생성된 공개키를 교환하는 알고리즘 동작 방법을 나타낸다.



(그림 1) ECDH 동작 과정

그림 1에서 클라이언트와 서버는 같은 타원곡선 E, 곡선의 범위를 나타내는 소수 P, 곡선상의 점 G를 공유한다. 클라이언트는 난수 c를 생성하고 곡선상의 임의의 한 점 G를 선택하여 공개키(cG)를 생성한다. 생성된 공개키는 서버(상대 단말)에게 전송된다. 마찬가지로 서버에서도 난수 k와 임의의 점 G를 이용하여 공개키를 생성하고 클라이언트에 공개키(kG)를 전송한다. 클라이언트와 서버는 서로 전송받은 공개키를 각각 자신이 생성한 난수만큼 연산을 하는데, 결과적으로 각각 연산한 값 즉, c(kG)와 k(cG)는 값이 같기 때문에 클라이언트와 서버는 같은 값을 가지게 되는 셈이다. 후에 생성된 비밀키를 이용하여 데이터를 암호화 및 복호화하게 된다.

3. ECDH 알고리즘의 문제점

ECDH 알고리즘에서 공개키를 생성하기 위해서는 공유 값 즉, 타원곡선 E, 소수 p, 타원곡선을 결정하는 상수 a, b와 임의의 좌표 G를 상대 단말에 전송하여야 한다. 위의 값들을 전송하기에 앞서 해당 파라미터들을 이용하여 타원곡선 E의 좌표인 G를 생성해야 하는데, G 값을 생성하기 위해 x 또는 y에 일정한 난수를 대입하여 타원곡선 식에 일치하는 값을 구해야 한다. 그러나 타원곡선 상의 상수 값(a, b)의 크기가 클수록 좌표를 생성하기 위한 시간이 오래 걸릴 수 있다. 위 과정 및 타원곡선 E에 기반한 각 파라미터 값들의 전송이 늦어진다면 단말간의 암호화를 위한 비밀키 생성 과정이 지연될 수 있다. 따라서 공개키를 만드는 과정 중의 하나인 좌표를 생성 및 설정하기 위한 빠른 알고리즘이 필요하다.

4. 제안하는 키 생성 알고리즘

ECDH 알고리즘에서 공개키를 생성하기 위하여, 좌표 G는 유한체 GF(p)에서 a, b ∈ GF(p)에 의해 정의된 타원곡선 위에 존재하여야 한다. 타원곡선의 범위이자 체의 표수인 P가 3 이상일 때 타원곡선의 식은 다음 식(1)과 같다.

$$y^2 = x^3 + a*x + b \qquad (1)$$

따라서 타원곡선 식에 수를 하나씩 대입하여 식에 맞는 값을 구하여야 한다. 기존의 공개키 (G)를 구하는 생성 알고리즘은 그림 2와 같다.

```
// p : 타원곡선 E의 범위, 소수
// x, y : 좌표G의 (x, y)
// a, b : 타원곡선 E를 결정하는 상수
for x from 0 to p-1 by +1
BEGIN
  for y from 0 to p-1 by +1
  BEGIN
   // l_side : 타원곡선 식의 왼쪽 변
   l_side <- y^2 % p ;
   // r_side1, 2 : 타원곡선 식의 오른쪽 변
   r_side1 <- x * x * x ;
   r_side2 <- a * x ;
   // r_sideSum : 오른쪽 변의 합
   r_sideSum <- r_side1 + r_side2 + b ;
     IF l_side == r_sideSum % p THEN
        PRINT x, y
  END
END
```

(그림 2) 기존의 공개키(G) 생성 알고리즘

제안한 키 생성 알고리즘은 좌표 생성 방법을 간단하게 하여, 수를 하나씩 대입하는 과정을 생략할 수 있다. 지정한 타원곡선 E가 x축의 0을 지나게 되면 구하고자 하는 G 값을 단순화 할 수 있다. 타원곡선 E가 x축의 0을 지나게 되면, 위의 식에 x=0을 대입하여 다음과 같은 연산 식 (2), (3), (4)로 계산할 수 있다.

$$y^2 = 0^3 + a*0 + b \qquad (2)$$

$$y^2 = 0 + 0 + b \qquad (3)$$

$$\therefore\ y = \pm\sqrt{b} \qquad (4)$$

따라서 G의 값은 x=0, y=±√b가 된다. 이 때 y는 b의

제곱 값이므로, b의 값을 이용하면 y를 쉽게 구할 수 있다. 제안한 공개키(G) 생성 알고리즘은 그림 3과 같다.

```
// p : 타원곡선 E의 범위, 소수
// x, y : 좌표G의 (x, y)
// a, b : 타원곡선 E를 결정하는 상수
x <- 0 ;
l_side <- y^2 % p ; //타원곡선 식의 왼쪽 변
r_side <- b % p ; //타원곡선 식의 오른쪽 변
// IF l_side == r_side 이므로 remove ' % p '
IF y^2 == b THEN
  PRINT x, y
```

(그림 3) 제안된 공개키(G) 생성 알고리즘

결과적으로, G의 좌표는 (0, ±√b)가 된다. 이 규칙을 적용시키면 각 단말에서 G를 생성하는 연산 과정을 줄일 수 있을 뿐만 아니라, G를 전송하지 않아도 되기 때문에 G를 전송함으로써 발생되는 연산 횟수가 줄고, 처리 지연 등의 문제가 나타나지 않는다.

5. 결론

최근 스마트폰 등의 모바일 단말이 활성화됨에 따라 모바일 어플리케이션을 이용한 서비스가 증가하고 있다. 모바일 어플리케이션을 통해 전송되는 데이터는 인터넷망을 이용하기 때문에 제 3자가 데이터를 수집할 수 있어 데이터의 암호화가 필요하다.

암호화 방식에는 암호화하는 보안의 강도가 강한 ECC 암호 알고리즘이 효율적이다. ECC 암호화 알고리즘은 비밀키 생성을 위해 ECDH 알고리즘을 이용하여 공개키를 교환해야 하는데, 단말에서 생성한 난수와 타원곡선위의 한 좌표 G를 이용하여 공개키를 생성해야 한다. 그러나 보안의 강도를 높이기 위해 파라미터 값들이 커지게 되면 공개키를 생성하는 시간이 오래 걸리게 되어 제약적인 모바일 환경에서는 비효율적이다. 따라서 본 논문에서는 공개키를 생성하는 요소인 타원곡선위의 점 G를 빠르게 생성할 수 있는 알고리즘을 설계하여 모바일 환경에서도 ECC 알고리즘을 효율적으로 사용할 수 있도록 하고자 하였다. 향후 기존 EDCH 키 교환 알고리즘과 제안한 공개키 교환 알고리즘을 측정 및 테스트를 통해 성능을 검증하고, 이를 통하여 제안한 ECDH 키 교환 알고리즘을 적용할 수 있는 프로그램을 구현할 것이다.

ACKNOWLEDGMENT

본 연구는 경원대학교의 지원으로 수행되었음

참고문헌

[1] 김갑열 외 2, "타원곡선 공개키 생성을 위한 고속 스칼라곱 연산 시스템 구현", 해양정보통신학회 2010. 2

[2] 고훈, "타원곡선 알고리즘을 이용한 XML 문서 암호 구현", 한국인터넷정보학회 8권 1호

[3] http://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

[4] W. Diffie and M. Hellman, "New Directions in Cryptography", IEEE Information Theory Workshop

[5] K. Kaabneh and H. Al-Bdour, "Key Exchange Protocol in Elliptic Curve Cryptography with No Public Point", American Journal of Applied Sciences 2 2005 Science Publications