# 타원곡선과 다중서버를 이용한 키로밍 프로토콜

문성원°, 김영갑, 문창주
고려대학교 컴퓨터학과
{kdunkman°, ygkim, mcj}@software.korea.ac.kr

# A Key Roaming Protocol using Elliptic-Curve and Multi-Server

SungWon Moon°, YoungGab Kim, ChangJoo Moon
Dept. of Computer Science and Engineering, Korea University

요 약

어플리케이션 시스템을 사용하려는 사용자에 대한 인증에 관한 문제는 네트워크에 접근하고자 하는 사용자가 실제 의도한 상대인지를 판단하는 것으로써, 서버에서 서비스를 제공하기 전에 만족해야 할 필수적인 요구조건이다. 이와 관련한 많은 연구들이 진행되어 왔는데, 이 중에서 패스워드를 이용한 사용자 인증 프로토콜은 여타의 방법에 비해 사용자에게 편리함을 제공할 뿐만 아니라 추가적인 하드웨어를 필요로 하지 않는 다는 장점을 가지고 있다.

최근에는 다중서버를 이용한 프로토콜이 제안되어 기존의 프로토콜 비해 보안이 강화된 서비스를 제공할 수 있게 되었다. 그러나 이 프로토콜은 다중서버를 사용함으로 인해서 부하가 많이 걸린다는 문제점이 존재한다. 본 논문에서는 이러한 다중서버를 이용한 프로토콜을 기반으로 하고, 이에 타원곡선 알고리즘을 적용함으로써 컴퓨팅 파워를 줄일 수 있는 방안을 제시하고자 한다.

## 1. 서 론

패스워드(password)를 기반으로 한 키 교환 프로토콜에 관한 연구는 1990년부터 시작되었으며, 그 목적은 인간이 기억할 수 있는 패스워드와 같은 정보를 이용하여 사용자를 인증하고자 하는 것이다. 여타의 방식이 사용자를 인증하기 위해서 별도의 하드웨어나 보조기억장치를 필요로 하는 반면에, 패스워드 프로토콜은 사용자의 패스워드만을 이용하여 사용자를 안전하게 인증할 수 있다는 장점을 가지고 있다. 그러나 사전공격(dictionary attack)과 같이 패스워드 자체가 가지고 있는 취약성으로 인한 보안상 문제점이 존재한다.

이를 해결하기 위한 초기 프로토콜([1],[2],[3])은 네트워크 상에서 클라이언트와 서버 간에 교환되는 메시지 자체에 대한 보안이 연구의 중심이었다. 최근에 진보된 형태의 프로토콜은 중앙서버에 대한 사전공격으로 인한 패스워드의 취약성에 대해 관심을 두게 되었으며 그 대표적인 예로는 베리사인(Verisign)의 포드(Ford)와 칼리스키(Kalisky)가 제안한 프로토콜[4]과 한국정보보호진흥원에서 제안한 프로토콜[5]이다. 이들은 중앙서버에 저장되어 있는 패스워드의 취약성을 보완하기 위한 프로토콜로써 다중서버를 이용하여 강한키를 획득한다.

본 논문에서는 다중서버(multi-server)를 이용한 프로토콜을 기반으로 하고, 이에 키의 크기(size)가 작으면서도 일반적인 공개키 알고리즘과 같은 보안을 제공할 수 있는 타원곡선 알고리즘을 적용하여 계산의 오버로드를 줄일 수 있는 새로운 프로토콜을 제안한다.

## 2. 관련연구

### 2.1 다중서버를 이용한 패스워드 프로토콜

패스워드 프로토콜의 초기 형태인 DH-EKE나 SPEKE와 같은 프로토콜은 클라이언트와 서버 사이에 패스워드를 공유하고 있다고 가정한 후, 이를 이용하여 강한키를 생성하는 프로토콜이다. 이 프로토콜은 네트워크 상에서 교환되는 패스워드에 대한 off-line 공격에 대응하기 위한 것으로써, 서버 자체에 대한 공격에는 매우 취약하다는 단점이 존재한다.

베리사인에 소속되어 있던 포드와 칼리스키는 기존의 단일서버를 이용하는 프로토콜이 가지고 있던 단점을 해결하기 위해서 다중서버를 이용하는 패스워드 프로토콜을 제안하였으며 이는 베리사인의 키로밍 서비스의 기본이 되고 있다. 베리사인의 패스워드 프로토콜은 크게 두 가지로 구성되어 있는데 키등록 프로토콜과 키로밍 프로토콜이다. 그림 1은 이 프로토콜의 키로밍 프로토콜을 나타내는 그림이다.

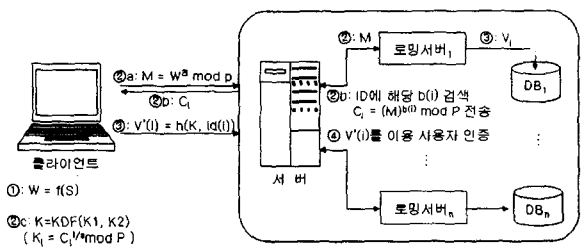


그림 1 베리사인의 키로밍 프로토콜

그림 1에서 볼 수 있는 것처럼, 이 프로토콜의 가장 큰 특징은 다중서버를 사용한다는 것과 패스워드에 대한 정보가 시스템의 어느 곳에도 저장되지 않는다는 사실이다. 이로 인해 패스워드에 대한 정보는 서버 측에게 공개하지 않음으로써 중앙 서버에 대한 사전공격에 대해서

대응할 수 있는 방안을 최초로 제안하였다. 그러나 이 프로토콜은 키를 로밍하는 동안에 별도의 보안채널을 필요로 하고 기존의 RSA 알고리즘의 이산대수문제를 이용하기 때문에 프로토콜이 진행되는 동안에 부하가 많이 걸리게 되며, 최근에는 이를 해결하기 위한 연구가 활발히 진행되고 있다.

### 2.2 타원곡선 알고리즘

공개키 알고리즘은 평문(plain-text)을 암호화하고 복호화하는데 있어서 공개키와 비밀키를 사용하는 알고리즘을 말하며 비대칭키 알고리즘이라 불리기도 한다. 오늘날 안정적이라는 평가를 받으면서도 많이 사용되는 알고리즘으로는 RSA(Rivest-Shamir-Adleman), ElGamal, ECC(Elliptic Curve Cryptography) 등이 있으며, 이 중에서 가장 많이 사용되는 알고리즘은 RSA이다.

하지만 RSA 알고리즘은 암호화의 안정성에 필요한 키의 크기가 점차 증가하고 있는 추세이다. 이로 인해서 알고리즘을 계산하는 과정에 있어서 발생되는 부하가 증가되어 암호화 및 복호화를 많이 요구하는 전자 상거래와 같은 시스템에는 적당하지 못하다.

RSA 이후에 등장한 ECC 알고리즘 같은 경우에는 RSA 알고리즘에 비해 훨씬 작은 길이의 키로 비슷한 강도의 보안성을 제공할 수 있을 뿐만 아니라 적은 크기의 비트 수의 추가로도 더 높은 보안성을 제공할 수 있어서 계산의 부담을 줄이는데 유용하다. 그 결과 최근에는 이 알고리즘을 적용할 수 있는 방안에 관한 연구뿐만 아니라 ECC가 가지고 있는 약점을 조사하는 연구가 활발히 진행되고 있다.

기존의 RSA 알고리즘을 사용하는 패스워드 프로토콜이 이산대수문제(discrete logarithm problem)를 사용하는 반면에 ECC를 패스워드 프로토콜에 적용하기 위해서는 타원곡선 이산대수문제(ECDLP: Elliptic Curve Discrete Logarithm Problem)를 기반으로 한다. 본 절에서는 이를 위해서 ECC의 매개변수를 결정하며 새로운 프로토콜을 제안할 때 이 제한조건을 전제적인 가정으로 두었다.

- 시스템 매개변수(System parameters)

1. 체의 크기(field size) $q$: 소수의 승수. 실제로는 $q = p$이거나 $q = 2^m$인 경우에 대해 구현된다.
2. 체(field) $F_q$의 두 원소 $a$와 $b$: $a$와 $b$는 $F_q$상의 타원곡선 E의 방정식을 결정하게 된다.
   (p>3인 경우: $y^2 = x^3+ax+b$,
   p=2인 경우: $y^2+xy = x^3+ax^2+b$)
3. 체 $F_q$의 두 원소 $x_p$와 $y_p$: 점 $P=(x_p,y_p)$는 $E(F_q)$에서 소수 위수 (prime order)를 갖는 한 점이다.
   ($P \neq O$이며 여기서 $O$는 무한원점(point at infinity))
4. 점 P의 위수(order) n.

- 시스템 매개변수 확인과정

시스템 매개변수 $(q,a,b,P = (x_p, y_p),n)$는 위의 조건을 만족시킬 수 있도록 하기 위해서 다음의 검증 과정을 거친다.

1. $q$가 소수의 승수 (prime power)임을 검증한다.
2. $a$, $b$, $x_p$, $y_p$가 $F_q$의 원소임을 확인한다.
3. $a$, $b$에 대한 타원곡선이 비특이(non-singular)임을 확인한다.
   ($p>3$인 경우: $4a^3 + 27b^2 \neq 0$, $p=2$인 경우: $b\neq 0$)
4. 점 P가 타원곡선 E의 한 점인 것과 $P\neq O$ 임을 확인한다.
5. $n>4\sqrt{q}$이고, $n$이 소수이고, $n$이 충분히 큰 소수 ( e.g., $n>2^{160}$ )임을 확인한다.
6. $nP = O$임을 확인한다.
7. 타원곡선의 특정 부류 (special class)에 대해 알려진 공격들을 피하기 위해, 모든 $1\leq k\leq 20$에 대해 $n$이 $q^k-1$을 나누지 않음과 $n\neq q$임을 확인한다.

## 3. 새로운 프로토콜

본 논문에서 제안하는 새로운 프로토콜은 RSA 공개키 알고리즘을 사용하는 다중서버를 이용하는 패스워드 프로토콜을 ECC로 대체하여 키의 크기를 줄일 수 있도록 한다. 그 결과 프로토콜 진행 동안에 발생하는 연산의 부하를 줄일 수 있어 더 작은 컴퓨팅 파워를 가지고 프로토콜을 진행할 수 있는 새로운 프로토콜을 제안한다.

### 3.1 프로토콜 매개변수

- S : 패스워드
- $E_q(a,b)$ : a, b, q를 파라미터로 가지는 타원곡선, 이 때, q는 소수이거나 $2^m$의 형태를 가진다.
- G : 키 교환을 위해 선택되는 기저점(basepoint). 위수가 충분히 큰 값(n)이어야 한다.
- $1 \leq a \leq q-1$ : 사용자가 선택하는 난수 (숨은 요소(blinding factor))
- $1 < b(i) < q-1$ : 각 i번째 로밍서버의 비밀정보
- KDF($K_1$, ... ,$K_2$) : 키 파생 함수
- OWF(K, j) : 일방향 함수

### 3.2 프로토콜 과정

기존의 다중서버를 이용하는 프로토콜과 마찬가지로 비밀정보를 등록하는 키등록 과정과 비밀정보를 다운받는 키로밍과정의 2단계 과정으로 구분할 수 있다.

1) 키등록 과정

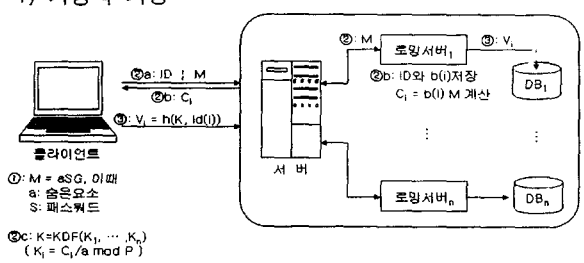


그림 2 새로운 프로토콜의 키등록 과정

① 프로토콜의 시작은 로밍요청정보(M)를 생성하는 것을 시작으로 한다. 이를 위해 클라이언트와 서버가

공유하고 있는 기저점 G를 사용자의 패스워드(S)와 숨은 요소(a)를 이용해서 암호화한다.

② 사용자 로밍요청정보에 대한 로밍응답정보를 생성하고 전송한다.

a. 생성한 로밍요청정보를 사용자의 ID와 함께 중앙 서버로 전송한다.

b. 각 로밍서버는 임의값 b(i)를 생성하고 사용자 아이디와 함께 저장한 후, 로밍응답정보($C_i$)를 생성하여 서버 측으로 전송한다.

c. 클라이언트 측에서는 전송받은 로밍응답정보를 이용하여 세션에서 사용할 키 값 K를 생성한다.

③ K를 이용하여 V(i)값을 생성한 후에 서버 측으로 전송한다. 각 로밍서버는 전송 받은 V(i)값을 자신의 로컬 데이터베이스에 저장한다.

2) 키로밍 과정

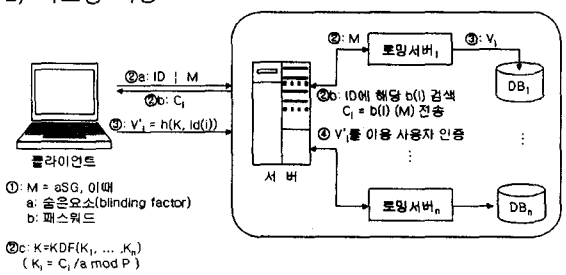


그림 3 새로운 프로토콜의 키로밍 과정

키로밍 과정은 키등록 과정과 매우 비슷한 과정을 거치게 되나 등록과정이 사용자 인증정보를 저장하는 과정인 반면에 로밍과정은 저장되어 있는 인증정보를 이용하여 실제 사용자를 인증하는 과정을 말한다.

① 등록과정과 마찬가지로 키로밍요청정보를 생성하는 것을 시작으로 한다.

② 로밍요청정보를 전송받은 서버는 로밍응답정보를 생성하여 클라이언트 측으로 전송한다.

a. 생성한 로밍요청정보를 사용자의 ID와 함께 중앙 서버로 전송한다.

b. 각 로밍서버는 사용자의 ID를 이용해서 데이터베이스에 저장되어 있는 임의값 b(i)를 검색하고 로밍응답정보($C_i$)를 생성하여 클라이언트 측으로 전송한다.

c. 클라이언트 측에서는 전송받은 로밍응답정보를 이용하여 세션에서 사용할 키 값 K를 생성한다.

③ 클라이언트 측에서는 ②의 과정에서 생성된 K를 이용하여 $V_i$값을 생성한 후에 서버 측으로 전송한다. 각 로밍서버는 다시 로컬 데이터베이스에 저장되어 있는 $V_i$값을 비교함으로써 사용자 인증과정을 마친다.

## 4. 비교 및 평가

다중서버를 기반으로 한 베리사인의 패스워드 프로토콜은 RSA 알고리즘을 기반으로 하고 있다. 그러나 현재 보안 기준을 만족시키기 위해서는 RSA의 경우에 1024 비트의 크기의 키를 사용해야 한다. 이에 반에, ECC 알고리즘의 경우에는 160비트 크기의 키로도 보안 조건을 충분히 만족시켜 줄 수 있다. 뿐만 아니라, 표 1에서 볼 수 있는 것처럼, ECC의 경우에는 더 높은 보안 수준을 요구할 경우에 RSA에 비해 더 적은 크기의 비트 추가로도 보안성을 만족시켜줄 수 있다.

표 1 RSA와 ECC의 보안성 비교

| Pollard-rho를 이용한 분석시간 (ECC) | | GNFS를 이용한 정수분해시간 (RSA) | |
|---|---|---|---|
| Key Size | MIPS-Years | Key Size | MIPS-Years |
| 150 | $3.8\times10^{10}$ | 512 | $3\times10^{4}$ |
| 205 | $7.1\times10^{18}$ | 768 | $2\times10^{8}$ |
| 234 | $1.6\times10^{28}$ | 1024 | $3\times10^{11}$ |
| | | 1280 | $1\times10^{14}$ |
| | | 1536 | $3\times10^{16}$ |
| | | 2048 | $3\times10^{20}$ |

## 5. 결론 및 향후연구

단일서버를 이용하는 패스워드 프로토콜은 네트워크 상에서 패스워드를 안전하게 전송함으로써 사용자를 인증할 수 있는 방안에 중심을 두고 있었다. 최근의 연구 방향은 중앙에 저장되어 있는 패스워드 자체에 대한 사전 공격에도 안전한 프로토콜인 다중서버를 이용하는 프로토콜에 대한 연구가 활발히 진행되고 있다.

베리사인에서는 다중서버를 이용하는 프로토콜을 제안하였으나 이 프로토콜은 몇 개의 로밍서버를 사용하기 때문에 단일서버에 비해 키를 교환하기 위한 부담이 큰 편이다. 본 논문에서는 베리사인의 프로토콜을 기반으로 하고 이 프로토콜이 발생시키는 부하를 줄이기 위해서 암•복호화 과정 동안에 RSA 알고리즘 대신에 타원곡선 알고리즘을 적용함으로써 프로토콜의 효율성을 높일 수 있는 방안을 제시하였다.

향후 연구로는 본 논문에서 제시한 프로토콜의 수학적인 검증이 필요하다. 또한, 이를 적용한 시스템을 개발하고 기존의 프로토콜을 적용한 시스템과 비교함으로써 그 효율성의 검증이 이루어져야 할 것이다.

## 참고문헌

[1] S. Bellovin and M. Merritt, "Encryped Key Exchange: Password-based Protocols Secure against Dictionary Attacks", Proc. IEEE Symposium on Research in Security and Privacy, May 1992.

[2] D. Jablon, "Strong password-only authenticated key exchange", ACM Computer Communications Review, October 1996.

[3] R. Perlman and C. Kaufman, "Secure Password-Based Protocol for Downloading a Private Key", Proc. 01999 Network and Distributed System Security Symposium, Internet Society, January 1999.

[4] W.Ford and B. Kaliski, "Server-Assisted Generation of a Strong Secret from a Password", Proc. 9th International Workshops on Enabling Technolcgies: Infrastructure for Collaborative Enterprises, IEEE, June 14-16, 2000.

[5] 김지연, "서버의 사전 탐색 공격을 고려한 패스워드 기반의 사용자인증 프로토콜", 대한민국특허, 2002년04월10일

[6] 이용기, 이정규, "타원곡선을 이용한 안전한 패스워드 프로토콜", 정보보호학회논문지, 9권 1호, 1999