타원 곡선 암호의 EMV 적용에 관한 연구

A Study on the Application of Elliptic Curve Cryptography to EMV

김웅* 임동진**
Woong Kim* Dong-Jin Lim**

Abstract - EMV was formed in February 1999 by Europay International, MasterCard International and Visa International to manage, maintain and enhance the EMV Integrated Circuit Card Specifications for Payment Systems as technology advances and the implementation of chip card programs become more prevalent. The formation of EMV ensures that single terminal and card approval processes are developed at a level that will allow cross payment system interoperability through compliance with the EMV specifications. A credit card environment of the domestic market adopted the standard Local-EMV to have the compatibility with EMV international standard and the EMV migration have been carried out by the step-by-step process. It may be possible to adopt various kinds of cryptographic algorithms, however, RSA public key algorithm is currently used. In this paper, as a public key algorithm for the authentication process, Elliptic Curve Cryptographic algorithm is applied to the EMV process. Implementation results is shown, and the possible changes necessary to accommodate Elliptic Curve Cryptography is proposed.

Key Words : EMV, elliptic curve cryptography, ECC, IC card, smart card

1. 서 론

EMV는 1999년 2월에 유로페이(Europay), 마스터카드(MasterCard), 비자(Visa)에 의해 지불시스템과 IC카드를 위한 국제적인 IC카드 표준의 제정, 유지 및 관리를 위하여 만들어 졌다. EMV의 등장으로 터미널과 카드 승인 프로세스들이 EMV 표준 규격하에 국제적인 호환성을 가지고 개발되어지고 있다.[1] 국내 신용카드의 경우 국제적으로 통용되는 EMV와 호환을 원칙으로 하는 로컬 EMV 표준을 채택함을 금융 감독원에서 밝히고 단계별로 시행하고 있다.

EMV 표준의 보안 프로세스는 3가지로 정적 데이터 인증(SDA), 동적 데이터 인증(DDA) 그리고 오프라인 핀 암호화(off-line PIN encryption)가 그것이다. 정적 데이터 인증은 발행사에 의해 할당되고 IC 카드에 저장되어진 데이터가 변경되지 않았음을 터미널이 확인하는 보안 프로세스이며, 동적 데이터 인증은 IC카드의 정적 데이터에 대한 인증을 수행한 후 카드에 의해서 발생된 서명에 대한 인증을 수행하는 프로세스이다. 이 EMV 인증 과정에 사용될 수 있는 암호 알고리즘은 여러 가지가 있을 수 있으나, 현재는 RSA 공개키 알고리즘이 사용되고 있다.

본 논문에서는 인증과정에서 사용될 수 있는 공개키 알고리즘의 한가지로 타원곡선(Elliptic Curve) 공개키 암호 알고리즘을 EMV 표준에 적용한다. 타원곡선 알고리즘을 EMV에 적용하여 구현한 결과와 이를 토대로 RSA가 아닌 다른 암호 알고리즘을 EMV가 수용하기 위하여 요구되어지는 EMV 표준의 변화를 제안한다.

저자 소개
* 김 웅 : 한양大學交 전기전자제어계측學科 碩士課程
Email: pince0@hotmail.com
** 임 동 진 : 한양大學交 전기전자제어계측學科 副敎授
Email: limdj@hanyang.ac.kr

2. 타원곡선 공개키 암호 알고리즘

체(field) F_P에서의 타원곡선군은 유한개의 원소를 가지고 있어 암호에 적용하기에 좋고 라운드 오차가 발생하지 않는다. 반면에 실수 타원곡선군에서 원소를 계산하는 것은 매우 느리고 라운드 오차 때문에 정확하지도 않다. 따라서 실제 암호에서는 유한체 F_P나 F_2^m위에서의 타원곡선을 사용한다. F_P가 체이려면 p가 소수이어야 한다. 이 체 F_P에서의 타원곡선은 a와 b의 값에 따라 여러 형태가 있고, F_P에 속한 숫자들의 순서쌍 (x, y)으로 표현되는 점들로 이루어지며 이러한 순서쌍을 더하거나 빼는 등 계산을 할 때에는 modulo p 연산을 사용한다.

F_p에서의 타원곡선 방정식은

$$y^2 \bmod p = x^3 + ax + b \bmod p$$

로 표현된다. $x^3 + ax + b$가 중복점을 가지고 있지 않다면 $(4a^3 + 27b^2 \bmod p \neq 0)$ 타원곡선 $y^2 \bmod p = x^3 + ax + b \bmod p$은 타원곡선을 만드는데 사용할 수 있으며, 타원곡선위의 점들과 무한 원점 O로 집합을 이룬다.

타원곡선군은 더하기 연산을 가지고 있는 군이다. 타원곡선에서는 어떤 점을 P라 하면 역원 -P를 구하면 항상 타원곡선위에 있게 된다. P, Q가 어느 한 타원곡선위의 서로 다른 점이라고 하면, 두 점을 더하기 위해 먼저 두 점을 연결하

는 직선을 그린다. 이 직선은 타원곡선과 반드시 한 점 -R에서 만나게 되는데 이 점 -R을 x축으로 대칭이동한 점 R이 P와 Q를 더한 점이다. 대수적으로 표현하면,

$$P+Q=R$$

$$s=\frac{y_P-y_Q}{x_P-x_Q}\bmod p$$

$x_R=s^2-x_P-x_Q \bmod p$, $y_R=-y_P+s(x_P-x_R)\bmod p$

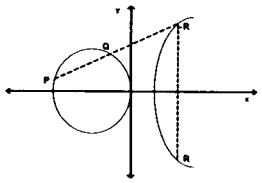
이다. 이때 s는 점 P, Q를 잇는 직선의 기울기이다. 2P를 구하기 위해서는 $y_P\neq 0$이라 하면,

$$2P=R$$

$$s=\frac{3x_P^2+a}{2y_P}\bmod p$$

$x_R=s^2-2x_P \bmod p$, $y_R=-y_P+s(x_P-x_R)\bmod p$

이다. 이때 a는 타원곡선에서 일차항 x의 계수이고, s는 점 P와 Q를 잇는 직선이다.



<그림 1> 타원 곡선

타원곡선 이산대수 문제는 두 원소 P와 Q에 대하여 kP=Q인 k를 찾아내는 문제이다. k값을 찾기 위한 하나의 방법은 점 P를 계속 더하여 몇 번 만에 점 Q가 되는지를 알아보면 되나 실제 암호시스템에서는 이 k가 매우 큰 수 이므로 이런 방법으로 k를 구하는 것은 힘들게 된다.

타원곡선 암호의 장점은 군(group)을 이루는 다양한 타원곡선을 활용할 수 있어 다양한 암호시스템 설계가 용이하다. 또 이 군에서의 subexponential time 알고리즘이 존재하지 않으므로 안전한 암호시스템을 설계하는데 있어 용이하다. RSA의 1024비트 키와 타원곡선 암호의 160비트 키가 같은 안전도를 갖을 정도로 타원곡선 암호는 다른 암호 시스템에 비해 더 작은 키 길이로 같은 안전도를 제공한다. 더욱이 타원곡선에서의 더하기 연산은 유한체에서의 연산을 포함하므로 하드웨어와 소프트웨어로 구현하기에 용이하다.

3. 타원곡선을 이용한 EMV 암호 프로세스 구현

타원곡선 파라메타의 생성을 위한 함수는 유한체를 정의하는 소수 p를 최소 160비트 이상의 크기로 생성하므로 유한체 F_P상의 타원곡선의 안전성을 확보해야 한다. 타원곡선 방정식 $y^2 \bmod p = x^3+ax+b \bmod p$를 위한 변수 a, b, 기본점 G 그리고 기본점 G의 위수 n의 경우에는 우선 x^3+ax+b가 중복점을 가지지 않도록 $4a^3+27b^2 \bmod p \neq 0$이어야 하며, Menezes와 Frey가 제시한 MOV 알고리즘 공격을 피하기 위해 $p^B \not\equiv 1 \bmod n\ (1\leq B\leq 30)$을 만족해야 한다. 또 위수 n 또한 160비트 이상인 소수이고 $n>4\sqrt{p}$ 이어야 한다. 타원곡선의 위수가 유한체의 크기와 같은 비정규 타원곡선의 경우 다항식 시간에 풀릴 수 있으므로 반드시 타원곡선 E의 위수를 확인하여 비정규 곡선을 제외 시켜야 한다.

타원곡선 키 쌍 생성 함수는 난수나 의사난수를 사용하여 통계적으로 유일하고 예측이 불가능한 정수 d를 (1, n-1)에서 선택하고, n을 법으로 하여 d의 역원 d^{-1}를 계산한다. ($dd^{-1}\equiv 1 \bmod n, 1\leq d^{-1}<n$). $Q=(x_Q,y_Q)=d^{-1}G$를 계산하여 키 쌍 (Q, d)을 취한다. 여기서 공개키는 Q, 비밀키는 d이다.

서명 생성 함수는 난수값 k를 생성하여 타원곡선 위의 점 $(x_1,y_1)=kG$를 계산하고 $r=x_1 \bmod n$을 계산하여 r=0이면 처음 단계부터 다시 시작한다. $k^{-1} \bmod n$을 계산하고 $e=k^{-1}\{h(m)+dr\} \bmod n$을 계산하여 $e=0$이면 처음 단계부터 다시 시작한다(h() : SHA-1). $t=d(k-e) \bmod n$을 계산하여 그 결과를 바이트열 s로 변환하고 메시지 M에 대한 서명으로 {r, s}를 출력한다.

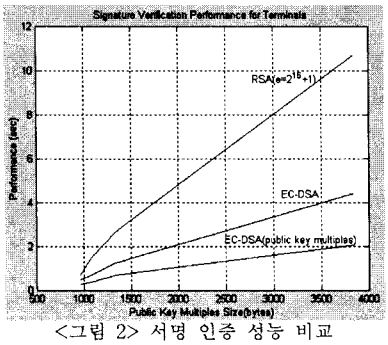
수신측에 수신된 이 서명에 대한 검증을 수행하는 함수는 수신된 서명의 첫 부분 r'의 비트길이가 해쉬(hash)함수의 출력 길이와 같은지 검증하고, 수신된 서명의 두 번째 부분 s'을 정수로 변환한 t'가 0<t'<n인지 검증한다. 수신된 서명의 한 부분 s'과 n을 법으로 하여 $e'=s^{-1} \bmod n$과 $h(m)$을 계산하여 중간값을 계산한다. $u_1=h(m)e' \bmod n$과 $v=x_0 \bmod n$을 계산하여 수신된 r'과 v가 같은지를 검사하여 수신된 서명을 검증하게 된다.[3]

이상과 같은 조건들을 만족하여 EMV의 보안 프로세스 중 정적 데이터 인증을 구현하였다. 정적 데이터 인증 방법은 우선 발행사는 인증기관의 개인키를 이용하여 서명되어진 발행사 공개키와, 발행사 개인키를 이용하여 서명되어진 어플리케이션 데이터를 IC 카드에 저장한다. 터미널은 IC 카드의 어플리케이션 데이터가 변경되지 않았음을 확인하기 위해, IC 카드에서 어플리케이션 데이터, 발행사의 공개키 인증서, 그리고 어플리케이션 데이터의 디지털 서명을 읽어온다. 이제 터미널은 터미널에 존재하는 인증기관 공개키를 이용하여 발행사 공개키를 되찾고 어플리케이션 데이터의 발행사 서명을 확인하기 위하여 이 발행사 공개키를 사용하게 된다.

구현 결과 타원곡선 암호가 RSA에 비해 우수한 성능을 나타내었으나 타원곡선 암호와 기존 RSA의 성능 비교는 실제 구현에 있어서 소프트웨어의 환경적 요소나 운영상의 차이 등으로 인해 직접적인 비교는 힘들다. 하지만 일반적인 암호 시스템 비교 결과를 배경으로 타원곡선 암호와 RSA를 비교해 보면, 일단 안전성에 있어서는 모듈로 p에서의 소인수분해 문제인 RSA는 Sub-exponential time 알고리즘이고 타원곡선 암호는 완전지수복잡도(fully exopnential time) 알고리즘으로 일반적으로 완전지수복잡도 알고리즘이 Subexponential time 알고리즘보다 어려운 문제로 알려져 있으므로 타원곡선이 RSA보다 안전성에 있어 더 우수함을 알 수 있다.

공개키와 비공개키를 계산하는 데에 드는 계산량의 경우는 암호화 및 검증의 속도를 판가름하는 중요한 요소로 같은 길이의 키를 사용한다고 할 때 타원곡선 암호가 RSA보다 10배 정도 빠르다고 알려져 있다. RSA 경우 수행속도를 높이기

위해서는 공개키의 길이를 줄이는 방법밖에 없는데 이 경우 안전성에 악영향을 주기 때문에 타원곡선에 비해 큰 키길이를 가지며 따라서 계산속도도 떨어지게 된다. 다음 <그림2>는 서명인증시 RSA와 타원곡선의 계산 성능을 나타낸다.[2]



<그림 2> 서명 인증 성능 비교

키 쌍과 파라메타의 저장에 소요되는 크기와 암호화된 메시지의 크기등에서도 타원곡선 암호는 RSA와 확연히 구별되어지는데 <표1>에서 그 성능을 나타낸다. (bits)

	타원곡선	RSA
공개 파라메타	481	-
공개키 크기	161	1088
비공개키 크기	160	2048
서명 크기 (2000비트 메시지의 경우)	320	1024
암호화된 메시지 크기 (100비트 메시지의 경우)	321	1024

<표 1> 타원곡선과 RSA의 크기 비교

이상과 같이 타원곡선 암호는 계산량, 키크기, 암호화된 메시지의 크기 등에서 RSA보다 우수하여 높은 속도, 적은 전력 그리고 보다 적은 코드 길이로 구현이 가능하다.

4. 타원곡선 알고리즘 적용을 위한 EMV표준의 변화

본 절에서는 타원곡선 암호를 EMV 표준에 적용하기 위해서 EMV표준에 요구되어지는 변화를 고려해 본다. EMV 표준은 현재 RSA를 기준으로 작성되어져 있으므로 EMV 표준의 변화는 필수적이다. 타원곡선 암호를 EMV에 적용하기 위해서는 기존의 이산대수상에서의 함수들을 타원곡선상으로 변환하는 것 외에도 여러 가지의 변화가 필요한데 우선 해당 도메인내의 관련자(entity)들의 역할 변화를 들 수 있다.

인증기관은 타원곡선 파라메타를 생성하고 검증해야하고, 이를 이용하여 타원 키 쌍을 생성 및 검증하여야 하는 역할을 수행해야 한다. 타원곡선 파라메타는 유한체를 정의하는 변수들, 타원곡선을 정의하는 변수들, 그리고 타원곡선상의 순환군과 관련된 변수들로 이루어지며 모든 관련자들이 공유하는 공개정보이다. 인증기관은 발행사에 공개키 인증서를 발급하는 과정에서 타원곡선 파라메타와 함께 이를 이용하여 생성한 검증키의 유효성을 확인하여 안전성을 확보하고 타원곡선 디지털 서명 알고리즘으로 공개키 인증서를 서명한다.

발행사의 경우에는 IC카드의 인증을 위한 정적데이터 또는 IC카드 공개키 인증서를 서명하는데 사용되어질 발행사 키 쌍을 인증기관으로부터 얻게된 타원곡선 파라메타를 이용하여 생성하고 개인키를 사용하여 서명한다. 또한 인증기관으로부터 생성된 발행사 공개키 인증서를 검증하는데 사용되어질 인증기관 공개키를 인증기관으로부터 수신하여 안전하게 저장할 뿐만 아니라 공개키가 유효한지 타원곡선 파라메타를 이용하여 검증한다.

EMV의 인증서 포맷의 경우에는 공개키 알고리즘 식별자 항목이 새로이 부여되어져야 하며, Exponent, Modulus등이 RSA를 위한 필수 데이터 항목으로 타원곡선 암호를 위해서는 소용없어지지만 호환과정에서의 융통성을 고려해 패딩값으로 처리하도록 한다.(기존 패딩값 'BB'를 사용) EMV 표준에서도 문서의 여러 테이블들이 RSA 알고리즘에만 적용이 되므로 타원곡선 암호에 맞게 수정 또는 삭제되어져야 하며, 인정 알고리즘 리스트에 타원곡선 암호가 추가되어야 한다.

비공개 서명키의 경우 현재와 마찬가지로 유효기간을 두어 비공개 서명키가 서명에 사용될 수 있는 기간을 명확히 해야한다. 하지만 이러한 사용주기는 RSA와 다르게 타원곡선 암호의 사용으로 인한 키의 변화가 예상되므로 정책적으로 사용주기가 변화, 결정되어야 한다.

마지막으로 현재 거래인증의 네트웍 메시지는 DES MAC를 사용하여 길이가 8바이트로 고정되어있다. 하지만 타원곡선 디지털 서명 알고리즘은 최소 40바이트(안전한 보안 수준을 위해 160비트 이상이어야 한다. 160x2)의 서명 데이터 길이를 가지므로 심각한 문제를 야기시킬 수 있다. 따라서 EMV에서 타원곡선 디지털 서명을 거래인증의 네트웍 메시지에 사용하는 것은 바람직하지 못하다고 생각되며, 8바이트에 기초한 DES를 그대로 사용하는 것이 바람직하리라 본다.

5. 결론 및 추후 연구 과제

본 논문에서는 EMV표준에 타원곡선 암호를 적용하는 방법을 제시하고 EMV의 암호 프로세스 중 정적 데이터 인증을 타원곡선 암호를 이용하여 설계하였다. 이를 통해서 타원곡선 암호를 EMV에 적용하기 위하여 요구되어지는 EMV표준의 변화를 제시하였다.

추후 연구 과제로는 터미널과 IC카드가 어느 한 암호가 아닌 기존의 RSA와 앞으로의 타원곡선 암호 모두를 지원하게 되므로 기존 암호 시스템인 RSA와의 호환성에 있어서 어떠한 문제도 야기되지 않는 RSA와 타원곡선 암호가 공존하는 시스템에 관심이 요구된다.

참 고 문 헌

[1] EMVCo."*EMV2000 Integrated Circuit Card Specification for Payment System*" 2000. Version 4.0

[2] EMVCo."*EMV Elliptic Curve Technical Report*" 2001. Version 1.0

[3] ANSI X9.62 "*Public Key Cryptography For The Financial Services Industry : The Elliptic Curve Digital Signature Algorithm.*" 1998.

[4] Henna Pietilainen "*Elliptic Curve Cryptography on smart cards*" Helsinki Univ. Press, 2000.