International Journal of Information & Network Security (IJINS)
Vol.1, No.3, August 2012, pp. 216~222
ISSN: 2089-3299

Elliptic Curve Digital Signature Algorithm Using Boolean Permutation based ECC

F. Amounas* and E.H. El Kinani**
* R.O.I Group, Informatics Department Moulay Ismaïl University, Faculty of Sciences and Technics Errachidia, Morocco
** A.A Group, Mathematical Department Moulay Ismaïl University, Faculty of Sciences and Technics Errachidia, Morocco

Article Info

Article history:

Received July 02th, 2012
Accepted July 22th, 2012

Keyword:

Elliptic Curve Cryptography,
Discrete Logarithm,
Boolean Permutation,
DSA, ECDSA

ABSTRACT

The Elliptic Curve Digital Signature Algorithm (ECDSA) is the elliptic curve analogue of the Digital Signature Algorithm (DSA), where it is a digital signature scheme designed to provide a digital signature based on a secret number known only to the signer and also on the actual message being signed. Digital signature schemes reduce transmission costs, because the message is contained in the signature itself and no separate message and signature need be sent again. In this paper, we attempt to provide more secure digital signature scheme by using Boolean permutation based elliptic curve cryptography (ECC). In particular, we introduce a proposed to development the original ECDSA with more complexity. In our scheme, using ECC and then applying the Boolean permutation generates the signature. The use of Boolean permation will provide better performance in this regard.

Corresponding Author:

E.H. El KINANI
Mathematical Department
Moulay Ismaïl University,
Faculty of Sciences and Technics, Box 509 Errachidia, Morocco
E-mail: elkinani_67@yahoo.com

1. INTRODUCTION

The Digital Signature Algorithm (DSA) was specified in a U.S. Government Federal Information Processing Standard (FIPS) called the Digital Signature Standard (DSS). Its security is based on the computational intractability of the discrete logarithm problem (DLP) in prime-order subgroups of Z^*_p.

In 1985, the Elliptic Curve Discrete Logarithm Problem (ECDLP) was proposed independently as a new cryptographic scheme by koblitz [1] and Miller [2]. It is considered that the security of ECC is sufficiently proved. Since the ECDLP appears to be significantly harder than the DLP, the elliptic curve systems appears efficient than in conventional discrete logarithm systems.

The Elliptic Curve Digital Signature Algorithm (ECDSA) is the elliptic curve analogue of the DSA. ECDSA was first proposed by Vanstone in 1992[3] in response to NIST's (National Institute of Standards and Technology) request for public comments on their first proposal for DSS. In 2004, Tenz et al [4] introduced a new digital signature schema with recovery based on ECC. They utilised the characteristic of ECC and self-certified public key, which was first proposed by Girault in 1991. Rajaram and al [5] introduced knapsack based elliptic curve cryptography. Their algorithm involves a high degree of sophistication and complexity. Therefore, they proposed an algorithm for digital signature with recovery using Knapsack on ECC. Under

this situation, we provide a new scheme using Boolean permutation based ECC. In fact, Boolean permutations are used in various different areas and play an important role in the security of cryptosystems. Their properties are explained in many references (see e.g [6]).
In our previous works, we provide an example of the public-key cryptosystems based on ECC mechanism [7] and the implementation of elliptic curve cryptosystem using Tifinagh characters [8]. Furthere, we have provided a new mapping method based on non-singular matrices [9]. In fact, the transformation of the message into an affine point is explained. A transformed character is encrypted by ECC technique. In the present work, we propose a new algorithm to generate a data sequence based Boolean permutation. Next, we will extended this algorithm for digital signature scheme. The proposed method has two levels of authenticated encryption. First, one is based on elliptic curve signature and second one is applying a Boolean permutation for the signed message. The proposed method provides high security in this regard.
The remainder of this paper is arranged as follows: first we start with brief review of ECC in section 2. Next, we review the original ECDSA in section 3. In section 4 we propose a new digital signature scheme using Boolean permutation on ECC. The security of the proposed method is studied in 5. The paper is concluded in Section 6.

2. OVERVIEW OF ELLIPTIC CURVE CRYPTOGRAPHY

The mathematical basis for the security of elliptic curve cryptosystems is the computational intractability of the elliptic curve discrete logarithm problem (ECDLP). ECC is a relative of discrete logarithm cryptography.
An elliptic curve E over F_p is defined by an equation of the form:

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

where a, $b \in F_p$, and $4a^3 + 27b^2 \neq 0 \pmod p$, together with a special point O, called the point at infinity. The set $E(F_p)$ consists of all points (x, y), $x \in F_p$, $y \in F_p$, which satisfy the defining equation, together with O. We denote the curve by E/F_p.
It is well known that E/F_p with a binary operation, called addition of points and denoted +, is an abelian group with O as the identity element. We denote the group by $E(F_p)$.
The basic EC operations are point addition and point doubling. Elliptic curve cryptographic primitives require scalar point multiplication. Say, given a point P(x, y) on an EC, one needs to compute kP, where k is a positive integer. This is achieved by a series of doubling and addition of P (see e.g [10]).
The addition of points is defined as:
Let be $P=(x_1,y_1) \in E(F_p)$ then $-P=(x_1,-y_1)$. If $Q=(x_2, y_2) \in E(F_p)$, $Q \neq P$, then $P + Q = (x_3, y_3)$ with

$$x_3 = (t^2 - x_1 - x_2)$$
$$y_3 = (t\,(x_1 - x_3) - y_1)$$

where

$$t = \begin{cases} (y_2 - y_1)(x_2 - x_1)^{-1} & \text{if } P \neq Q \quad \text{(addition)} \\ (3x_1^2 + a)(2y_1)^{-1} & \text{if } P = Q \quad \text{(doublement)} \end{cases}$$

Therefore we apply doubling and addition depending on a sequence of operations determined for k. Every point x_3, y_3 evaluated by doubling or addition is an affine point (points on the Elliptic Curve).

3. THE ELLIPTIC CURVE DIGITAL SIGNATURE ALGORITHM (ECDSA)

The ECDSA is the elliptic curve analogue of the DSA. Digital signature schemes are the counterpart to handwritten signatures. A digital signature is a number that depends on the secret key only known by the signer and on the contents of the message being signed. Signatures must be verifiable without access to the signers private key. Signatures should be existentially unforgeable under chosen message attacks. This asserts that an adversary who is able to obtain Alices signatures for any messages of his choice cannot forge Alice signature on a single other message. In this section, we will introduce first the original ECDSA. Suppose Alice wants to send a digitally signed message to Bob. They first choose a finite field GF(p) where p is a prime, an elliptic curve E, defined over that field and a base point P with order n.

Alice's key pair is (d ,Q), where d is her private key and Q is her public key.

To sign a message M Alice does the following:

- ECDSA SIGNATURE GENERATING

Step 1. Chooses a random integer k with $1 \leq k \leq n-1$.

Step 2. Computes $kP = (x_1, y_1)$ and $r = x_1 \bmod n$. If $r = 0$ then she returns to step 1.

Step 3. Computes $k^{-1} \bmod n$.

Step 4. Computes $e = h^{-1}(M)$.

Step 5. Computes $s = k^{-1}(e + dr) \bmod n$. If $s = 0$ then she returns to step1.

Step 6. Alice signature for the message M is (r,s).

To verify Alice's signature (r,s) on the message M , Bob obtains an authenticcopy of Alice's parameters and public key. Bob should validate the obtained parameters, Bob then does the following:

- ECDSA SIGNATURE VERIFICATION

Step 1. Verify that r and s are integers in the interval [1, n-1].

Step 2. Compute $e = h^{-1}(M)$

Step 3. Computes $w = s^{-1} \bmod n$.

Step 4. Computes $u_1 = ew \bmod n$ and $u_2 = rw \bmod n$.

Step 5. Computes $X = u_1P+u_2P$. If X=O then he will reject the signature.

Otherwise compute $v = x1 \bmod n$ where $X=(x_1, y_1)$.

Step 6. Accept the signature if and only if v = r.

If the signature (r,s) on the message M was indeed generated by Bob, the $s = k^{-1}(e + dr) \bmod n$. With this information we have:

$$
\begin{aligned}
k &\equiv s^{-1}(e + dr) \bmod n \\
&\equiv (s^{-1}e + s^{-1}rd) \bmod n \\
&\equiv (we + wrd) \bmod n \\
&\equiv (u_1 + u_2d) \bmod n
\end{aligned}
$$

thus $(u_1P + u_2Q) = (u_1 + u_2d)P = kP$ and so v = r as required.

4. MAIN RESULTS

In this paper, the plaintext and signed symbols are points on EC. Each point are coded on m bits. It is known that a Boolean function of n variables is a mapping from an n-dimensional vector space over the binary field $F_2=\{0, 1\}$ to itself. Such a function can be implemented as a combinational logic unit with one bit output and n-bit input [11]. A mapping from F^n_2 to F^m_2 is called an (n,m) Boolean function. An (n,m) Boolean function can always be expressed as a collection of m functions in F_n, where F_n is the set of all Boolean functions over n variables. A particular class of these type of multiple output Boolean functions occur when m = n and that different inputs yield different outputs. By treating each input/output as the binary expression of a points Pi within the range $S1 = \{P_i, i = 0, 1, ... , n-1\}$, the above functions perform permutations on S1 and are called Boolean permutations [12].

Our procedure use such collection of Boolean functions over m variables called as Boolean permutation, BP, of order m:

$$BP = [f_1(x), f_2(x), ..., f_m(x)]$$

Like any permutation, the Boolean permutation BP has own inverse. The Boolean permutation and its inverse are defined by tables, as shown in Table.1and Table.2, respectively.

Table 1. Boolean Permutation BP

m bits binary representation of points P_i	BP				
	f_1	f_2	f_3	…	f_m
1010110001	1	0	1	…	1
1001000111	0	1	1	…	0
1101001100	0	1	1	…	1
…	.	.	.	…	.
1110010110	0	1	0	…	1
0000000001	1	1	0	…	1

Table 2. Boolean Permutation BP^{-1}

BP^{-1}	x_1	x_2	x_3	...	x_m
1010101101	1	0	1	...	1
0110101101	1	0	0	...	1
1110101101	1	1	0	...	0
...	.	.	.	...	.
0100111101	1	1	1	...	0
1100111101	0	0	0	...	1

The input to a table consists of m bits that represent a points on EC.

4.1. Algorithm (1) using the Boolean permutation

Input: P the base point on EC, n=order(P), points P_i on EC
Output: Seq

1. Consider the sequence of points P_i for i=0 to n-1.
2. Each point is coded on m bits.
3. Constructs a Boolean permutation of order n over m variables noted BP.
4. For each point P_i, search its binary form in BP, noted d.

$$S_i = BP(P_i) = BP(d)$$

5. The sequence formed is :

$Seq = \{S_0 = BP(P_0), S_1 = BP(P_1), ..., S_{n-1} = BP(P_{n-1})\}$

Return Seq

4.2. Proposed Algorithm Description

The proposed scheme is divided into two phases: signature generation phase and signature verification phase. Suppose Alice wants to send a digital signed message to Bob. They first choose a finite field GF(p), an elliptic curve E defined over that field and a base point P with order n. A point P and E are publicly known, as is the embedding system $M \rightarrow P_M$, which imbed plaintext on an elliptic curve E. In this proposition the original message M is represented by a point on EC.

Alice's private key is $k_A \in [1..n]$, $P_A = k_A P$ is her public key.

Signature generation phase

If Alice wants to sign a message M, she does the following:

Step1. Generate a data sequence Seq.

Step2. Chooses a random integer k, with $1 \leq k \leq n$ and compute

$$Q = kP_A$$

Step3. Compute the signature (r, s) of the message M with:

$$r = P_M + Q = (x, y), \text{ and } s = xQ,$$

if x=0 then return to step2. Here, r and s are a points on EC.

Step4. Apply the Boolean permutation value for the signed message.

$$R = Seq(r) \text{ and } S = Seq(s)$$

Then, send a signed message (R, S)

Signature verification phase

After receiving (R, S) Bob can apply the reversal of permutation to recover (r, s). To verify Alice's signature (r, s) on the message M which is not sent, Bob get Alice's parameters and her public key. Then Bob does the following:

Step 1. Extract a group of m bits in sequence of R (Similarly for S).

Step 2. Apply the reverse permutation BP^{-1}.

$$r = BP^{-1}(R) \text{ and } s = BP^{-1}(S)$$

Step 3. Verify that r and s are a point on EC and represent r = (x', y')

Step 4. Recover and verify the message M with $P_M = r - (x')^{-1}s$ where x' is the x-coordinate of r.

Step 5. The procedure is repeated for the next groups, which is not visited earlier.

then reverses the embedding to get back the message M.

If the signature (r,s) on the message M was indeed generated by Bob, s = xQ

Then: $k \equiv (x)^{-1}s \mod n$

$K \equiv (x)^{-1}(xQ) \mod n$

$\equiv ((x)^{-1}x)Q \mod n$

$\equiv Q$

$\equiv kP_A$

If K = O or $((x')^{-1}x) \neq 1$ then he refuses to accept this signature else $r - K = r - (kP_A) = P_M$ is a point on the elliptic curve and reverses the embedding to get back the message M.

Example: Illustration

Let E be an elliptic curve given by the following equation:

$$y^2 = x^3 - x + 16 \text{ [29]} \qquad (2)$$

The set of all points on the elliptic curve are:

{(5, 7), (28, 4), (18, 1), (22, 12), (6, 20), (13, 5), (2, 14), (21, 11), (23, 3),(10, 7), (14, 22) , (16, 23) , (7, 27), (1,4) ,(0,4), (0,25), (1,25), (7,25), (16,6), (14,7), (10, 22), (23,26), (21,18), (2, 15), (13, 24), (6, 9), (22, 17), (18, 28), (28, 25), (5, 22), O}

The points on the elliptic curve are shown below in Figure 1.

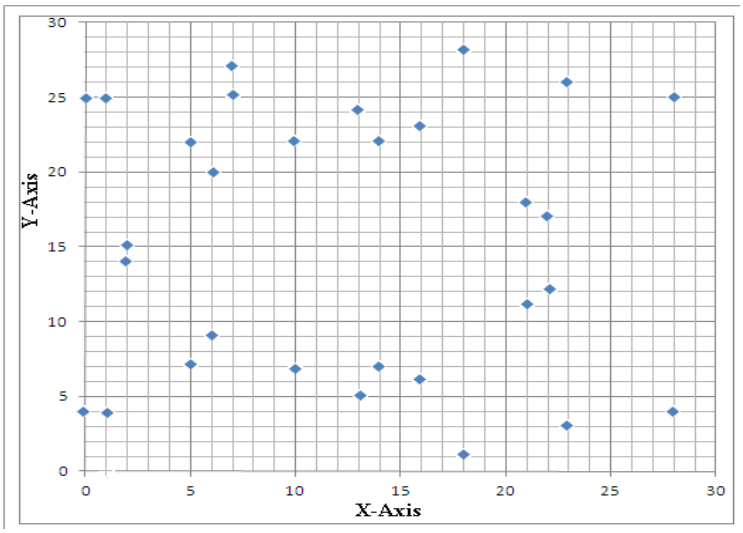

Figure 1. The elliptic curve $E_{29}(-1, 16)$

The base point P is selected as (5,7). Here the choosing curve contains 31points with P is the point generator. It is the point which represents the letter 'A', as well as 2P represents the letter 'B',..., 31P represents space.
In our case we use the letters 'A' to 'Z' with some of the other symbols like ';', ', ', '.', '?' and space for illustration purpose only.
Alice wants to sign a message "IJINS" to Bob. Then, she chooses a parameters such that:

k_A = 21 (Private key), and P_A=(10,22) is her public key.

k = 13 and Q = kP_A=(13,24).

The table below (table 3) shows the results for to sign the message "IJINS". The signature is generated and then verified by the values of r and s.

Table 3. The signed message

Character	P_M	r	s
I	(23,1)	(18,1)	(0,25)
J	(10,7)	(22,12)	(10,22)
I	(23,1)	(18,1)	(0,25)
N	(1,4)	(21,11)	(28,25)
S	(16,6)	(7,27)	(14,7)

Here, we choose Boolean permutation BP and BP^{-1} as following:

$BP(P_i)$={{1010101101},{0110101101},{1110101101},{0001101101},{1001101101}, {0101101101}, {1101101101}, {0011101101}, {1011101101}, {0111101101}, {1111101101}, {0000011101}, {1000011101}, {0100011101}, {1100011101}, {0010011101}, {1010011101}, {0110011101}, {1110011101}, {0001011101}, {1001011101}, {0101011101}, {1101011101}, {0011011101}, {1011011101}, {0111011101}, {1111011101}, {0000111101}, {1000111101}, {0100111101}, {1100111101}}

BP^{-1}={{0010100111}, {1110000100}, {1001000001}, {1011001100}, {0011010100}, {0110100101}, {0001001110}, {1010101011}, {1011100011}, {0101000111}, {0111010110}, {1000010111}, {0011111011}, {0000100100}, {0000000100}, {0000011001}, {0000111001}, {0011111001}, {1000000110}, {0111000111}, {0101010110}, {1011111010}, {1010110010}, {0001001111}, {0110111000}, {0011001001}, {1011010001}, {1001011100}, {1110011001}, {0010110110}, {0000000001}}

Therefore, the signed message is (R, S) with R = BP(r) and S = BP(s)

R= 11101011010001101101111010110100111011011000011101
S= 00100111011101011101001001110110001111010001011101

5. SECURITY ANALYSIS

In this section, we give an analysis about our proposed scheme. A number of attacks against the proposed scheme are presented. There are two basic attacks against public-key digital signature schemes: (1) Key-only Attacks: In these attacks, an adversary knows only the signer's public key, (2) Message Attacks: Here an adversary is able to examine signature corresponding either to known or chosen messages. Message attacks can be further subdivided into three classes, 1) known-message attack, 2) chosen-message attack, and 3) adaptive chosen message attack. In our proposed digital signature scheme with message recovery, we consider two kinds of attacks.

Attack 1. An adversary attempts to derive the user's private key kA from known public information (E, p, n and P point, P_A public key of user) available. We have shown this attack is not possible in our proposed scheme. In fact, an adversary can not derive $P_A = k_AP$ from known public information, because ECDLP to obtain Alice private key k_A is difficult.
Attack 2. Suppose an adversary attempts to decrypt the message M from the digital signature R, S without knowing user Alice's private key k_A in our proposed scheme.
Proof. In Attack 2, suppose an adversary attempts to decrypt the signed message M then he has to know the binary sequence Seq. This value is complex. So, the adversary's attempt of applying the reverse of permutation will not work. So, Attack 2 is not possible in our proposed scheme.

6. CONCLUSION

In this work, we provide a new digital signature scheme with message recovery using Boolean permutation based ECC. The signed message generated by applying the Boolean permutation is highly secured. The security of the proposed method is based on the difficulty of inverting Boolean permutation. The proposed scheme could successfully ward of these possible attacks and provide better performance in this regard. In our knowledge, this is the first work which proposes a new digital signature scheme based Boolean permutation.

REFERENCES

[1] Koblitz, Neal. Elliptic Curve Cryptosystem, *Mathematics of Computation*, vol 48, no, 177, pp 203-209, 1987.
[2] Miller. Victor S., Use of Elliptic Curves in Cryptography, Lecture Notes in Computer Sci.no. 218, pp. 417-426, *Springer-Verlag*. 1986.
[3] Vanstone, S. A., Responses to NISTs Proposal Communications of the ACM, 35, 50-52, 1992.
[4] S. F. Tzeng and M. S. Hwang, Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem, *Computer Standards and Interface*, vol. 26, no. 2, pp. 61-71, 2004.
[5] R. Rajaram, M. A. Prabakar, M. I. Devi, and M. Suguna, Knapsack based ECC encryption and decryption, *International Journal of Network Security*, vol.9, no.3, pp. 218-226, Nov. 2009.
[6] Fengrong Zhang, Yupu Hu, Min Xie, Juntao Gao and Qichun Wang, Constructions of Cryptographically Significant Boolean Permutations, *Applied Mathematics & Information Sciences*, vol.6, No. 1, 117-123 , 2012.
[7] F.Amounas, E.H. El Kinani and A. Chillali, An application of discrete algorithms in asymmetric cryptography, *International Mathematical Forum*, Vol. 6, no. 49, pp.2409-2418, 2011.
[8] F.Amounas and E.H. El Kinani, Cryptography with Elliptic Curve Using Tifinagh Characters, *Journal of Mathematics and System Science* Vol.2, No.2, pp.139-144, 2012.
[9] F. Amounas and E.H. El Kinani, Fast Mapping Method based on Matrix Approach For Elliptic Curve Cryptography, *International Journal of Information & Network Security (IJINS)* Vol.1, No.2, pp. 54-59, 2012.
[10] H. Lange and W. Ruppert, Addition laws on elliptic curves in arbitrary characteristics, Journal of Algebra, Vol.107(1),106-116, 1987.
[11] Chrystos H.Papadimitriou, *Computatonal Complexity*. (1994).
[12] Chuan-Kun Wu, Vijay Varadharajan. Public Key Cryptosystems Based on Boolean Permutations and their Applications. School *of Computing & Information Technology*, University of Western Sydnay (Nepean), PO Box 10, Kingswood, NSW 2747, Australia.

BIOGRAPHY OF AUTHORS

EL HASSAN EL KINANI received the Ph.D in mathematical physics in 1999 from Mohamed V University Rabat Morocco. He is full professor at department of mathematics in Moulay Ismaïl University, Faculty of Sciences and Technics, Errachidia, Morocco. He is interested in classical and quantum cryptography.

E-mail: elkinani_67@yahoo.com

FATIMA AMOUNAS received the DESS (diploma of high special study) degree in informatic in 2002 from Sidi Mohamed Ben Abdellah University, Faculty of Sciences Dhar El Mehrez, Fès Morocco. She is currently a Ph.D student in University Moulay Ismaïl, Faculty of Sciences and Technics, Errachidia, Morocco. Her research interests include elliptic curve and cryptography.

E-mail: F_amounas@yahoo.fr