New Elliptic Curve Digital Multi-Signature Schemes for Multi-Section Messages

Dang Minh Tuan
National Laboratory for Information Security
Hanoi, Vietnam
Email: dangtuan@vietkey.vn

Abstract—In this paper, we first present a new elliptic curve digital signature schemes, which are suitable for multi-signature. Then, we propose a new concept of multi-signature for multi-section messages, where any one of a group can sign different sections of a message.

Keywords-elliptic curve, multi-signatures, multi-section messages;

I. INTRODUCTION

Since first proposal by Victor Miller and Neal Koblitz in 1985, Elliptic Curve Cryptography (ECC) has evolved into a mature public-key cryptosystem. ECC offers the smallest key size, the highest strength per bit, and computational efficiency, so it can be used for both client devices and server machines. Many research works have been done on digital signature schemes based on ECC. The Elliptic Curve Digital Signature Algorithm (ECDSA) is the elliptic curve version of the Digital Signature Algorithm (DSA), and it is widely standardized elliptic curve-based signature scheme, appearing in the ANSI X9.62, IEEE 1363-2000, FIPS 186-2 and ISO/IEC 15946-2 [1][2].

A multi-signature scheme is the one that enables a group of signers to authenticate a document producing a fixed-length digital signature. The goal of a multi-signature is to prove that each member of the group is participated in signing the message.

Unfortunately, the ECDSA is not suitable for multi-signature schemes, hence some non-standard elliptic curve algorithms were developed as alternatives for multi-signing [3],[4],[5], [6], [7], or [8], [9],[10]. In this paper, we proposed a new elliptic curve digital signature schemes, then we applied it to multi-signature, and further more for signing multi-section messages.

There are some situations, when we need to use multi-signatures for multi-section messages, where each signer of the group can sign single or multiple parts of a message. For example, a company releases a document that may involve engineering department, financial department and human resources department. Each entity is responsible for preparing and signing one or more parts of the document.

One of the differences between our multi-signature scheme and other recently in use is that with our scheme, any user can sign more than one part of a message while with any other, one can be able to sign only a part of the message.

II. PRELIMINARY

A. Elliptic Curves

An elliptic curve is a algebraic curve defined by an equation of the form:

$$y^2 = x^3 + ax + b$$

(II.1)

Which is non-singular (has no cusps or self intersections). The curve is non-singular if the discriminant $4a^2 + 27b^2 ≠ 0$. This is called the Weierstrass equation for an elliptic curve.

We always take $a, b, x$ and $y$ to be elements of a field such as $\mathbb{R}, \mathbb{Q}, \mathbb{C}$, or finite field $\mathbb{F}_q$, where $q = p^n$ with $p$ prime and $n ≥ 1$. If $K$ is a field with $a, b ∈ K$, then the elliptic curve is said to be defined over $K$. The point $(x,y)$ on the elliptic curve with $(x,y) ∈ K$ is called a $K$–rational point.

For technical reasons, a point at infinity (Infinity) is added to an elliptic curve. Denote this point by $O$ sitting at the top and bottom of the $y$-axis. One of the most important properties of elliptic curves is the existence of a group law for adding points on the curve. Consider two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$; $P_1 ≠ P_2$ on the elliptic curve $E : y^2 = x^3 + ax + b$.

Define $P_3 = P_1 + P_2$, where $P_3 = (x_3, y_3)$ is the reflect point ($x$-axis) of $P_3'$ on the curve and $P_3', P_1, P_2$ are on the same line. As we shall see [11], the formulae for $P_3$ are as follows:

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2$$

(II.2)

$$y_3 = -y_1 + \left(\frac{y_2 - y_1}{x_2 - x_1}\right) \times (x_1 - x_3)$$

(II.3)

Theorem 2.1: For all $P, P_1, P_2, P_3$ on $E$, the addition of points on elliptic curve $E$ satisfies the following properties:

- Commutativity: $P_1 + P_2 = P_2 + P_1$;
• Existence of identity: \( P + O = P \);
• Existence of inverse: Given \( P \) on \( E \), there exists \( P' \) on \( E \) with \( P + P' = O \);
• Associativity: \( P_1 + (P_2 + P_3) = (P_1 + P_2) + P_3 \).

For details of the proof refer to [11].

B. Elliptic Curve Digital Signature Algorithm

In the following section, we briefly describe the digital signature algorithm based on elliptic curve.

Algorithm 1 ECDSA Signature Generation

INPUT: Domain parameters \( D = (q, \text{FR}, S, a, b, P, n, h) \), private key \( d \), message \( m \).
OUTPUT: Signature \( (r,s) \).
1: Randomly select \( k \in [1, n-1] \).
2: Compute \( R = kP = (x_1, y_1) \) and convert \( x_1 \) to \( \bar{x}_1 \).
3: Compute \( r = \bar{x}_1 \mod n \).
4: if \( r = 0 \) then
5: goto step 1;
6: end if
7: Compute \( e = H(m) \).
8: Compute \( s = k^{-1}(e + dr) \mod n \).
9: if \( s = 0 \) then
10: goto step 1;
11: end if
12: Return \( (r,s) \).

Algorithm 2 ECDSA Signature Verification

INPUT: Domain parameters \( D = (q, \text{FR}, S, a, b, P, n, h) \), public key \( Q = dP \), received message \( m' \), signature \( (r,s) \).
OUTPUT: Acceptance or rejection of the signature.
1: Verify that \( r \) and \( s \) are integers in interval \( [1, n-1] \). If not, then return("Reject the signature").
2: Compute \( e' = H(m') \).
3: Compute \( w = s^{-1} \mod n \).
4: Compute \( u_1 = e'w \mod n \) and \( u_2 = rw \mod n \).
5: Compute \( R' = u_1P + u_2Q \).
6: if \( R' = \infty \) then
7: return("Reject the signature").
8: end if
9: Convert \( x_1 \) of \( R' \) to an integer \( \bar{x}_1 \).
10: Compute \( r' = \bar{x}_1 \mod n \).
11: if \( r' = r \) then
12: return("Accept the signature").
13: else
14: return("Reject the signature").
15: end if

We denote \( H \) as a cryptographic hash function whose outputs have bitlength no more than that of \( n \) (the order of the point \( P \)). Suppose Alice wants to send a signed message \( m \) to Bob. Initially, the curve parameters \( (q, \text{FR}, S, a, b, P, n, h) \) must be agreed upon. \( q \) is the field size; \( \text{FR} \) (Field Represntation) is an indication of the basis used; \( a \) and \( b \) are two field elements that define the equation of the curve; \( S \) (Domain Parameter Seed) is an optional bit string that is present if the elliptic curve was randomly generated in a verifiable fashion; \( P \) is a base point of prime order on the curve; and \( h \) is the cofactor (where \( \#E(K) = h \cdot q \)).

Proof of signature verification:
If a signature \( (r, s) \) on a message \( m \) was indeed generated by the legitimate signer, then \( s \equiv k^{-1}(e + dr) \mod n \). We have to prove that if \( m' = m \) or \( e = e' \) then \( r = r' \). Indeed, \( R' = u_1P + u_2Q = (u_1 + u_2d)P = (e's^{-1} + rdr^{-1})P = k(e' + rd)(e + rd)^{-1}P \), and so \( R' = R \) or \( r' = r \) as required.

III. PROPOSED SIGNATURE SCHEME

A. Signature generation

We shall continue using the algorithm for ECDSA Signature Generation with the signature being computed as follows (at step 8):
\[
s = (k - ed)r^{-1} \mod n \quad \text{(III.1)}
\]
B. Signature verification

Using the Algorithm for ECDSA Signature Generation as shown above, we compute \( u_1, u_2 \) as follows (at step 4):
\[
u_1 = sr \mod n \quad \text{(III.2)}
\]
\[
u_2 = e' \mod n \quad \text{(III.3)}
\]
C. Proof of signature verification

\( R' = u_1P + u_2Q = srP + e'Q = (k - ed)r^{-1}P + e'P = kP = R \), if \( m' = m \) or \( e' = e \).

IV. NEW GENERIC MULTI-SIGNATURE SCHEME FOR MULTI-SECTION MESSAGES

We assume that there are \( t \) signers \( U_i \); \( 1 \leq i \leq t \) to sign a message \( m \in \{0, 1\}^* \). Let \( m \) divide into \( \ell \) sections, so we can express \( m \) in a form: \( m = (m_1 || m_2 || m_3 || \cdots || m_\ell) \). Let us define signing vectors \( \mathcal{V}_i = [\mathcal{V}_i[1], \cdots, \mathcal{V}_i[j], \cdots, \mathcal{V}_i[\ell]] \), where \( \mathcal{V}_i[j] = 1 \); \( 1 \leq j \leq \ell \) if signer \( U_i \) has to sign the section \( m_j \), and \( \mathcal{V}_i[j] = 0 \) if signer \( U_i \) does not have to sign the section \( m_j \). For the group of users, we have \( \mathcal{V} = (\mathcal{V}_1 || \mathcal{V}_2 || \cdots || \mathcal{V}_t || \cdots || \mathcal{V}_t) \).

A. Key generation

1) Each signer \( U_i (1 \leq i \leq t) \) randomly selects \( \ell \) integers \( d_i[j] \) as private keys from the interval \( [1, q-1] \); \( 1 \leq j \leq \ell \) and compute a corresponding public key as the point: \( Q_i = (\mathcal{V}_i \odot d_i)P \), where \( \mathcal{V}_i \odot d_i = \sum_{j=1}^{\ell} \mathcal{V}_i[j] \times d_i[j] \mod n \).
2) Group public key is equal to the sum of all individual public keys: \( Q = \sum_{i=1}^{t} Q_i \).
B. Signing phase

1) Each signer $U_i$ randomly selects $\ell$ integers $k_i[j]$ from interval $[1,q-1]$, $1 \leq j \leq \ell$, and computes $R_i = (\mathfrak{V}_i \otimes k_i)P$, which is broadcast to every other signer. Then one can compute the point $R = \sum_{i=1}^{t} R_i$, and converts $R = (x_R,y_R)$ to an integer as $r = x_R \pmod{n}$.

2) Each signer $U_i$ is responsible for preparing some sections of message $m$. He sends $H_i(m_j); 1 \leq j \leq \ell$ to the predetermined clerk, who then computes hash value for section $m_j$ as follows: $e_j = \sum_{i=1}^{t} (\mathfrak{V}_i[j] \times H_i(m_j)); 1 \leq j \leq \ell$ and $e = H(e_1 \parallel e_2 \parallel \cdots \parallel e_{\ell})$. By the way how to compute the hash value of the message, we will get multi-signature with distinguished signing authorities.

3) Let us define functions: $s_i = s_i(k_i,e,r,d_i) \pmod{n} ; s = \sum_{i=1}^{t} s_i \pmod{n} ; u_1 = u_1(s,e,r) \pmod{n} ; u_2 = u_2(s,e,r) \pmod{n}$ so that $u_1P + u_2Q = R$. Each signer $U_i$ computes $s_i$ and transmits $(R_i,s_i)$ to the clerk $C$. The clerk should verify the validity (evidence) of the individual signature $(R_i,s_i)$. If all of the individual signatures are legal, then the clerk generates the multi-signature $(\mathfrak{V},r,s)$ by computing $s = \sum_{i=1}^{t} s_i \pmod{n}$.

C. Signature verification

1) Compute $e'_j = \sum_{i=1}^{t} (\mathfrak{V}_i[j] \times H_i(m'_j)); 1 \leq j \leq \ell$ and $e' = H(e'_1 \parallel e'_2 \parallel \cdots \parallel e'_{\ell})$. Here $\mathfrak{V}_i$ is easily extracted from $\mathfrak{V}$.

2) Compute $u_1 = u_1(s,e',r) \pmod{n} ; u_2 = u_2(s,e',r) \pmod{n}$.

3) Compute $R' = u_1P + u_2Q$; $r = x_R \pmod{n}$.

4) If $r = r'$, the signature is valid.

D. Proof of signature verification

If $m = m'$ or $e = e'$ then $u_1 = u'_1$; $u_2 = u'_2$ so $u_1P + u_2Q = KP$ or $R' = R$.

V. PROPOSED MULTI-SIGNATURE SCHEME

A. Signing phase

• Each signer $U_i$ ($1 \leq i \leq t$) has to compute $R_i = (\mathfrak{V}_i \otimes k_i)P$; $R = \sum_{i=1}^{t} R_i$, and converts $R = (x_R,y_R)$ to an integer as $r = x_R \pmod{n}$.

• The signer $U_i$ computes $s_i = (\mathfrak{V}_i \otimes k_i - e(\mathfrak{V}_i \otimes d_i))r^{-1} \pmod{n}$ and transmits $s_i$ to the clerk. Then the clerk computes a solution $s = \sum_{i=1}^{t} s_i \pmod{n}$.

• The clerk checks the evidence of individual signature by computing: $R'_i = s_iP + eQ ; 1 \leq i \leq t$. If $R'_i = R_i$, the evidence is valid.

• Multi-signature is obtained in a form: $(\mathfrak{V},r,s)$.

B. Signature verification

• The verifier computes $u_1 = sr$; $u_2 = e'$.

• Next, he/she computes $R' = u_1P + u_2Q$; $r' = xR' \pmod{n}$.

• If $r = r'$, the signature is valid.

C. Proof of signature verification

$$R' = u_1P + u_2Q$$
$$= srP + e'Q$$
$$= \left(\sum_{i=1}^{t} s_i\right)P + \left(\sum_{i=1}^{t} e'_i \sum_{i=1}^{t} \mathfrak{V}_i \otimes d_i\right)P + e'\left(\sum_{i=1}^{t} \mathfrak{V}_i \otimes d_i\right)P$$
$$= \left(\sum_{i=1}^{t} \mathfrak{V}_i \otimes k_i\right)P + e\left(\sum_{i=1}^{t} \mathfrak{V}_i \otimes d_i\right)P$$
$$= R$$

VI. SECURITY ANALYSIS AND DISCUSSION

The security of the new multi-signature schemes for multi-section messages is based on the security of the underlying signature scheme. Hence the underlying signature scheme are based on the elliptic curve discrete logarithm problem, the member’s secret keys are secure.

Since each signer is responsible for preparing some sections of message, each signer may have distinguished signing responsibility. The new schemes provide additional evidence of each section of the message such that the members of the group can use it to prove their distinguished signing responsibility. Consider the security of the individual signatures $(R_i,s_i)$. Suppose that a attacker wants to forge signature $(R_i,s_i)$ on message $m$. First, the attacker could computes $R_i$, $r$, $e$, after that he/she should compute $s_i$ by solving the elliptic curve discrete logarithm problem $R_i = s_iP + eQ$, at the first scheme, or $R_i = (s_i + r)P + eQ$ at the second scheme. Therefore, it is hard to forge the $(R_i,s_i)$.

In order to successfully forge a group signature, the attacker should to know all signing keys. Consider the security of the group signature $(r,s)$. Suppose that the attacker wants to forge $(r,s)$ on a given $m$ or $e$; he/she may determine $r$ first and then after find $s$, or he/she may determine $s$ first.
and then find \( r \). Both cases lead to solving the elliptic curve discrete logarithm problem \( R = s r P + eQ \) at the first scheme or \( R = (s + r)P + eQ \) at the second scheme. Therefore, it is infeasible to forge \((r,s)\).

In case some verifiers only allowed to access partial contents of the message, the partial contents can still be verified using the group public key without revealing whole message. This feature can be achieved by just providing the one-way hash values of the inaccessible contents to the verifier.

The multi-signature is easily verified by using the group public-key without knowing the member’s public keys.

In case the signing procedures are fixed or remain unchanged, \( \mathcal{G} \) is predetermined and we can assume it as one of the parameters of multi-signature schemes.

At our multi-signature schemes, every signer can have every public key and private key for every section of the messages. That means the schemes are more secure than any other, but on the other hand, the computation and communication between signers is more complicated, so we could simplify the computation by choosing only a pair public-private key for all the sections of the messages.

If the number of the signers is the same as the number of sections of the messages \((t = \ell)\) and \( \mathcal{G}_i[j] = 1 \) for \( i = j \), and \( \mathcal{G}_i[j] = 0 \) for \( i \neq j \), then our multi-signature for multi-section becomes known multi-signature as the one in [5]. Further more, if \( t = \ell = 1 \) and proper choosing the function \( s_i \), then we shall have single-signature as ECDSA.

VII. CONCLUSION

It can be said that proposed multi-signature schemes for multi-section messages have high generality and they are very flexible in use and suitable for many practical applications.

ACKNOWLEDGMENT

The author gratefully acknowledges the many helpful suggestions of Ph.D Nguyen Hieu Minh during the preparation of the paper.

REFERENCES


