Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 1977798, 9 pages
https://doi.org/10.1155/2022/1977798

Research Article

An ECC-Based Digital Signature Scheme for Privacy Protection in Wireless Communication Network

Ping Zhang,[1] **Yamin Li**,[2] **Muhua Liu**,[1] **Youlin Shang**,[1] **and Zhumu Fu**[3]

[1] *School of Mathematics and Statistics, Henan University of Science and Technology, China*
[2] *School of Cyber Engineering, Xidian University, China*
[3] *College of Information Engineering, Henan University of Science and Technology, China*

Correspondence should be addressed to Yamin Li; 2390043823@qq.com

Received 27 February 2022; Accepted 22 March 2022; Published 13 April 2022

Academic Editor: Zhiguo Qu

Aiming at the security problems of wireless communication network and the shortcomings of Elliptic Curve Digital Signature Algorithm (ECDSA), this paper designed a forward secure digital signature scheme and proved that the scheme has forward security and unforgeability in the random oracle model. Experimental simulations are done in this paper, and the results show that the proposed scheme has the highest efficiency compared with the ECDSA scheme and the three existing forward secure schemes. This scheme not only meets the forward security and protects the users' privacies but also reduces the requirement of computing power of the user in the wireless communication network.

1. Introduction

In recent years, with the popularization of mobile devices in daily life, wireless network communication has developed rapidly. Now, for most mobile terminals, the wireless network is even the only means to access the network. The most prominent feature of the wireless network is that it breaks through the limitations of the wired network, saves a lot of line maintenance costs, and spreads more widely and flexibly. As an extension of the wired network, the application of the wireless network has greatly improved the efficiency of users' work and life [1–3]. However, with the deepening of wireless network applications, the security problems of wireless networks are becoming more and more obvious. The potential security risks of wireless networks pose a serious threat to users' information security. The security requirements of wireless network communication include the following three aspects. (1) Confidentiality: the wireless network communication should ensure the information security of both sides of the communication and prevent others from illegally using the information. (2) Integrity: the wireless network communication should ensure the integrity of data information to prevent illegal users from modifying, inserting and deleting information unintentionally or maliciously. (3) Nonrepudiation: the wireless network communication needs to realize the nonrepudiation of the information between the two communicating parties, so as to prevent the sender from denying that it has sent the information after sending the information, and also prevent the receiver from denying that it has received the information after receiving the information. In the above security requirements, the confidentiality is guaranteed by encryption technology, while the integrity and nonrepudiation need to use digital signature technology. Digital signature is the key technology in privacy protection scheme.

The digital signature plays an important role in many occasions, such as identity authentication, data integrity protection, and anonymity. It can solve many problems, such as signature forgery, repudiation, impersonation, and tampering [4]. The wireless communication network is a restricted environment, i.e., time (key generation, signature and verification, etc.) and space (data memory, program memory, bandwidth, code and data length, etc.) limit the achievement of security goals. In addition, wireless devices have relatively low computing power and relatively small storage space. This limits the application of cryptography technology, which requires a large number of complex operations, in wireless communication network. Therefore, how to design an efficient and secure digital signature scheme for wireless communication networks has become an urgent problem.

The common digital signature includes RSA digital signature, ElGamal digital signature, and elliptic curve (ECC-based) digital signature. When the key lengths of the three digital signatures are equal, the ECC-based digital signature [5] has the highest security. In 1985, Neal Koblitz [6] and V.S. Miller [7] proposed this algorithm, respectively. In the ECC-based digital signature, it is the elliptic curve finite group [8], instead of the finite cyclic group. Compared with the former two digital signatures, the ECC-based digital signature has the following advantages. Firstly, there are many different elliptic curves in the same finite field, which additionally ensures security. Secondly, the ECC-based digital signature is based on ECDLP, which is more difficult to solve than the discrete logarithm problem on the multiplication group of the prime field [9]. Finally, the key length of the ECC-based digital signature is shorter when the same level of security is required. In 1992, Scott Vanstone firstly proposed ECDSA, which was defined as a standard digital signature algorithm by the International Standards Organization [10]. However, in the elliptic curve digital signature, the leakage of the key will bring great loss to the users.

Anderson firstly proposed forward security to prevent key leakage in CCCS'97. However, he only put forward a brief description and did not give a specific scheme. Two years later, Bellare and Miner [11] not only proposed a detailed concept of forward security in Crypto'99 but also designed a forward secure digital signature scheme firstly. Forward security of the digital signature means that an attacker cannot know the key of previous time and forge the signature of previous time even if he acquires the key of a certain time [12]. The main idea of this technology is easy. That is, the system time is divided into many periods so that the key is different at any period [13]. Therefore, in a forward secure digital signature scheme, even if the private key of a certain period is leaked, the security of the message will not be affected in the previous period. This paper designs a forward secure ECC-based digital signature scheme for privacy protection in wireless communication network and proves that the scheme has forward security and unforgeability in the random oracle model. Compared with the ECDSA scheme and three existing forward secure schemes, the improved scheme not only meets the forward security and protects the users' privacies but also reduces the requirement of computing power of the user in the wireless network.

1.1. Related Works. In 2000, based on any regular scheme (such as RSA and DSA), Krawczyk [14] proposed a more efficient and simple digital signature scheme with forward security. In the same year, Abdalla and Reyzin [15] improved Bellare's scheme [11] in the ROM. They have improved the scheme with the shorter keys, which increased the practicability of this scheme. Bellare introduced some methods to construct digital signature schemes with forward security. Abdalla and Reyzin summarized and supplemented Bellare's methods. In 2001, Malkin et al. [16] combined the existing schemes into a new forward secure digital signature scheme without knowing the total number of periods. This scheme not only can take any digital signature scheme as the underlying module but also does not depend on any specific assumptions. They also constructed the first efficient digital signature scheme with forward security, which does not need to determine the total period in advance. In the same year, Itkis and Reyzin [17] designed a digital signature scheme with forward security, which requires only four modular exponential with short exponents in signature and verification. They proved this scheme is secure in the ROM. In 2002, Kozlov and Reyzin [18] designed a digital signature scheme that only needs a simple modular square in the KeyUpdate. They proved this scheme is secure based on the Fiat-Shamir transformation and the strong RSA assumption. In 2003, Fei et al. [19] designed a new digital signature scheme based on bilinear mapping, which has strong robustness. They proved this scheme is secure based on the computational Diffie-Hellman assumption. In 2004, McCullagh and Barreto [20] proposed a new efficient digital signature scheme with forward security. In the same year, Kang et al. [21] designed two digital signature schemes using the same KeyUpdate algorithm. In 2006, Boyen et al. [22] firstly proposed the forward security digital signature with un-trusted updates.

In 2011, Buchmann et al. [23] designed a hash-based digital signature scheme with a smaller signature size. In 2011, Liu et al. [24] proposed a ring signature scheme with forward security. This scheme ensures that all previous signatures containing this member are valid even if the key of some ring member is disclosed. In 2012, Yao-Chang et al. [25] analyzed the shortcomings of the digital signature in Electronic Medical Record (EMR) and proposed a forward secure digital signature scheme for EMR. In 2013, based on the Guillou-Quisquater signature scheme and Rabin cryptosystem, Guang-bao et al. [26] proposed a strong forward secure digital signature scheme. In 2014, based on Schnorr's digital signature scheme and Shamir's (t, n) threshold scheme, Yao-Chang et al. [27] proposed an efficient group signature scheme with forward security. This scheme has integrity and improves the efficiency of authentication of EMR in the KeyUpdate. In 2015, Zhenping et al. [28] proposed an ID-based forward secure digital signature algorithm using the Chebyshev public-key algorithm. The public key of this algorithm is the identity information of the signer. In addition, this algorithm has higher security, which based on large integer factorization and Cheyshev discrete logarithm problem. In 2016, based on the elliptic curve, Keyuan [29] proposed a digital signature scheme with message recovery, which not only resists forgery signature attack but also is forward secure. In the same year, Yarong et al. [30] proposed a forward secure proxy signature scheme, which is secure for all requirements of forward secure proxy signature. Based on ElGamal system, Shun-bo et al. [31] designed a digital signature scheme by using one-way hash chain technology. In 2017, based on the elliptic curve, Jinyuan and Xianghua [32] designed a forgery signature method to solve the security problem of the digital signature scheme with forward security. In 2017, for embedded real-time systems, considering the limited sensor resources and time constraints, Kim et al. [33] designed a forward secure digital signature scheme. In the ROM, they proved the proposed scheme is secure and gave the concrete implementation of the scheme. In the same year, aiming at the shortcomings of the certificateless signature scheme, Xu et al. [34] firstly designed a forward secure certificateless digital signature scheme based on random lattice in the standard model. They proved the strong unforgeability of the scheme based on small integer solution problem. In 2019, Xiaoping [35] combined certificateless public-key system with forward security and proposed a proxy blind signature scheme. This scheme solved the problem of key escrow and certificate management. Based on the above works, we propose a forward secure elliptic curve digital signature scheme to enhance the security of information systems.

1.2. Contribution. Aiming at the security problems of wireless communication network and the shortcomings of ECDSA scheme, this paper constructed an improved digital signature scheme and proved the security of the scheme. Compared with the ECDSA scheme, the improved scheme added key update algorithm to achieve the forward security and avoided the modular inversion by adjusting the signature formula. Compared with the ECDSA scheme and three existing forward secure schemes, the improved scheme reduces the requirement of computing power of the user in the wireless network. So, it is more suitable for wireless communication network.

1.3. Organization. We organized our paper as follows. The first section is the introduction of this paper. The second section introduces the knowledge of quadratic congruence equation, elliptic curve cryptography, and ROM. In the third section, we introduce the formal definition of Forward secure digital signature scheme (FSDSS), including adversary model and security definition. In the fourth section, we design an ECC-based digital signature scheme (the improved scheme). In the fifth section, we prove the improved scheme is secure. In the sixth section, we compare the efficiency of the improved scheme with the other four schemes. Finally, in the seventh section, we summarize the full text.

2. Preliminaries

2.1. Quadratic Congruence Equation. The general form of the quadratic congruence equation is

$$ax^2 + bx + c \equiv 0 \mod N, a \equiv 0 \mod N, \tag{1}$$

which can be transformed to another equation:

$$x^2 \equiv a' \mod N, \left(a', N\right) = 1. \tag{2}$$

This form is the standard form of the quadratic congruence equation.

When the decomposition of N is known, it is simple to calculate x. However, when the decomposition of N is unknown, it is difficult to calculate x. Therefore, the quadratic congruence equation has the same difficulty as the factorization problem [36]. The theory of quadratic congruence equation will be used in the KeyUpdate of the improved scheme.

2.2. Elliptic Curve Cryptography. The elliptic curve (cubic smooth algebraic curve) can be expressed as Weierstrass equation

$$y^2 + axy + by = x^3 + cx^2 + dx + e. \tag{3}$$

In this equation, a, b, c, d, e are constants defined in the field. If the characteristic of the field is not 2, equation (3) can be transformed into

$$\left(y + \frac{ax}{2} + \frac{b}{2}\right)^2 = x^3 + \left(c + \frac{a^2}{4}\right)x^2 + \left(d + \frac{ab}{2}\right)x + \left(e + \frac{b^2}{4}\right), \tag{4}$$

which can be written as

$$y_1^2 = x^3 + a_1 x^2 + a_2 x + a_3, \tag{5}$$

with $y_1 = y + (ax/2) + (b/2)$ and some constants a_1, a_2, a_3. If the characteristic of the field is also not 3, we can let $x_1 = x + (a_1/3)$ and obtain

$$y_1^2 = x_1^3 + Ax_1 + B, \tag{6}$$

for constants $A, B(\Delta = 4A^3 + 27B^2 \neq 0)$.

In the real field, in addition to all points on the elliptic curve, the Abel addition group also includes the infinity point O (zero point).

Definition 1. (Abel addition group of elliptic curve). The points of the elliptic curve E can generate an Abel addition group, if the addition of points on E satisfies the following five properties.

(i) (Closure): on E, for any points P_1 and P_2, there is a point P_3 with $P_1 + P_2 = P_3$

(ii) (Associativity): for any points P_1, P_2, and P_3 on E, there is $(P_1 + P_2) + P_3 = P_1 + (P_2 + P_3)$

(iii) (Existence of identity): for any point P on E, there is $P + O = P$

(iv) (Existence of inverses): on E, for any point P, there is a point P' with $P + P' = O$. This point P' is often denoted as $-P$

(v) (Commutativity): for all points P_1 and P_2 on E, there is $P_1 + P_2 = P_2 + P_1$

Definition 2. (Elliptic curve addition rule). The elliptic curve addition rule is defined as follows.

(i) (Add to zero point): let O be the unit element of the addition. For any point R on E, there is $R + O = R$

(ii) (Add to the inverse element): for any point $R = (x, y)$ on E, its additive inverse is $-R = (x,-y))$. There is $(-R) + R = R + (-R) = R - R = O$

(iii) (Add two points): for any points $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ on E, there is $P + Q = (x_3, y_3) = (\lambda^2 - x_1 - x_2, \lambda(x_1 - x_3) - y_1)$, where

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & \mathrm{P} \neq \mathrm{Q} \\ \dfrac{3x_1^2 + a}{2y_1} & \mathrm{P} = \mathrm{Q} \end{cases}. \tag{7}$$

Definition 3. (Elliptic curve discrete logarithm problem). For an elliptic curve E, G is a point on E, and its order n is a prime number. For any random number k, it is easy to calculate $Q = kG$. However, if G and Q are known, it is very difficult to find k.

2.3. Random Oracle Model. In ROM, the random oracle is a deterministic, accessible publicly, random uniform distribution function. It uniformly selects a value with deterministic length from the output field and makes the value as the answer (output) for the query (input) of the message with any length.

In the scheme, the hash function is idealized as a random oracle. The adversary can only obtain the required hash value by asking the random oracle. Between random oracle and adversary, there is also a simulator in the model. The simulator transforms the ability of the adversary into an advantage of breaking a known difficult problem.

In the ROM, the provable security is regarded as a simulation game. It is called training that the simulator answers all of the queries defined by the adversary. At the end of the simulation game, if the adversary can complete the pre-determined challenge from the simulator, the simulation game succeeds. This predetermined challenge contains some knowledge. The simulator can use this knowledge to solve the difficult problem in the scheme. Therefore, if the success probability of the simulation game is a nonnegligible value, this difficult problem is no longer difficult in the environment with the given adversary. This contradicts the computational difficulty of the known difficult problem in the real world.

3. Forward Secure Digital Signature Scheme

In the FSDSS, we assume that the system time is divided into T periods, and the key is different in every period. In the KeyUpdate, the private key SK_j of the j-th period can be calculated by the private key SK_{j-1} of the $(j-1)$-th period. However, it is difficult to calculate SK_{j-1} from SK_j. During a certain period, since the signer uses only the private key SK_j of this period, the key leakage of this period does not endanger the security of the system in the previous period.

Definition 4. (Forward secure digital signature). The FSDSS generally consists of the following four algorithms [37].

$\mathrm{Gen}(1^\lambda, \cdots) \longrightarrow (\mathrm{PK}, \mathrm{SK}_0)$: this is a random algorithm. This algorithm takes a security parameter λ and other system parameters as input and outputs the public key PK, the initial private key SK_0

$\mathrm{Update}(j, \mathrm{SK}_{j-1}) \longrightarrow \mathrm{SK}_j$: this algorithm takes the private key SK_{j-1} of the previous period as input and outputs the private key SK_j of this period

$\mathrm{Sign}(j, \mathrm{SK}_j, m) \longrightarrow S_j$: this algorithm takes the private key SK_j of the j-th period and the message m as input and outputs the signature S_j of the j-th period

$\mathrm{Vrfy}(\mathrm{PK}, m, S_j) \longrightarrow \{0, 1\}$: this algorithm takes the public key PK, the message m, and the signature S_j of the j-th period as input. If the verification is valid, this algorithm outputs 1, otherwise 0. When $\mathrm{Vrfy}(\mathrm{PK}, m, S_j) = 1$, the signature S_j is the valid signature of message m at the j-th period

3.1. Adversary Model. We assume that the adversary is a third party; that is, we only consider external attacks. The adversary's goal is to forge the signature information of the unsigned message. We assume that an adversary can submit the query of any message to the random oracle. After the adversary knows the result from the random oracle, he can still query the message.

3.2. Security Definition. Assuming that FSDSS is a four tuple $FSDSS = (Gen, Update, Sign, Vrfy)$, and $\mathscr{A}$ is an adversary. Next, let us consider the following game as Algorithm 1.

Definition 5. (Existential unforgeability). If no PPT adversary can win this game with a nonnegligible advantage, the signature scheme FSDSS = (Gen, Update, Sign, Vrfy) is existentially unforgeable under an adaptive chosen-message attack. Let $\mathrm{Succ}_{\mathscr{A}} = \Pr\left[\mathrm{game}(\lambda, \mathscr{A}, \mathrm{FSDSS}) = 1\right]$. In this paper, "PPT" represents probabilistic polynomial time.

4. Construction

In 1999, ECDSA became an ANSI standard. In 2000, ECDSA became an IEEE and NIST standard. However, there are three double-point arithmetic, four modular multiplications, and two modular inverses in the ECDSA scheme. Therefore, the operation of ECDSA is complicated, and the ECDSA scheme does not have the forward security. Once the attacker gets the private key, he can forge the signatures of all messages of the previous periods. The purpose of this paper is to design an improved ECC-based digital signature scheme with forward security. In the improved scheme, we add the key update algorithm to ensure the forward security of the scheme. In the

```
$game(\lambda, \mathscr{A}, \mathrm{FSDSS})$
$\{(PK, SK_0) \longleftarrow \mathrm{Gen}(1^\lambda)$
breakin$(j^+)(1 \leq j^+ < T)$: $SK_{j^+} \longleftarrow \mathrm{Update}(\cdots \mathrm{Update}(SK_0))$
Sign$(j, m)(1 \leq j < j^+)$: $S_j \longleftarrow \mathrm{Sign}(SK_j, m)$
$(j^*, m^*, S_j^*) \longleftarrow (\mathscr{A}\lambda, PK|\mathrm{breakin}(j^+), \mathrm{Sign}(j, m))$
If $PK$ is invalid then return 0.
If $0 \leq j^* < j^+$, $m^* \notin M$ and $1 \longleftarrow \mathrm{Vrfy}(PK, m^*, S_j^{*\cdot})$
then return 1, else return 0
}
(Assuming that $M$ is the set of messages $m$ submitted by $\mathscr{A}$ to the signature oracle.)
```

Algorithm 1

signature generation algorithm and signature verification algorithm, we simplify the calculation to improve signature efficiency. The improved scheme is as follows.

4.1. Parameter Selection. Assuming that the system's time is divided into T periods, $GF(q)$ is a finite field with order q. $E: y^2 = x^3 + ax + b(\mathrm{mod}\ q)$ $(a, b \in GF(q), 4a^3 + 27b^2 \neq 0)$ is an elliptic curve over finite field $GF(q)$. G is the base point of the elliptic curve E with order n. That is, $ord(G) = n$, where n is a large prime number. N is a composite number $(N > n)$, $h = \#E(GF(q))/n$, and $(h \ll n)$ is the cofactor. $\#E(GF(q))$ represents the number of points on the elliptic curve E, which is defined on the finite field $GF(q)$. $H : \{0, 1\}^* \longrightarrow \{0, 1\}^{160}$ is a secure hash algorithm. $\{0, 1\}^*$ represents a string with any length. $SK_0 = a_0$ $(a_0 \in [1, n-1])$ is the initial private key, and $\mathrm{PK} = a_0{}^{-1}G$ is the public key. Then, we keep parameter SK_0 secret, expose elliptic curve parameters $D = \{q, a, b, G, n, h\}$, the public key PK, and hash function $H(\cdot)$.

4.2. Key Update. In the j-th period $(1 \leq j \leq T)$, the signer uses the key SK_{j-1} of the $(j-1)$-th period to calculate the key S $\mathrm{K}_j = \mathrm{SK}_{j-1}^2 \bmod N$ of the j-th period. Therefore, in the j-th period $(1 \leq j \leq T)$, the private key is $\mathrm{SK}_j = a_0{}^{2^j} \bmod N$, where $a_0^{2^j}$ is precalculated in each period. The key SK_{j-1} is deleted, and the key SK_j is saved in the secret key list.

4.3. Signature Generation. The signer calculates the signature of message m in the j-th $(1 \leq j \leq T)$ period [(1)].

(1) At first, select a random number $k \in [1, n-1]$

(2) Compute $k^{2^{-j}}G = (x_1, y_1)$, $r = x_1 \bmod N$. If $r = 0$, return to step (1)

(3) Compute the hash value $e_j = H(j, m, r)$

(4) Compute $s_j = \mathrm{SK}_j k - e_j \bmod N$. If $s_j = 0$, return to step (1)

(5) Get the signature $S_j = (r, s_j)$

4.4. Signature Verification. The verifier checks whether (r, s_j) is the signature of message m in the j-th period $(1 \leq j \leq T)$ after receiving the signature (r, s_j) and message m [(1)].

(1) At first, verify $r, s_j \in [1, n-1]$. If not, return directly and reject the signature

(2) Compute the hash value $e'_j = H(j, m, r)$

(3) Compute $w_j = e'_j + s_j \bmod N$

(4) Compute $X = w_j{}^{2^{-j}}PK = (x_2, y_2)$. If $X = 0$, reject the signature

(5) Compute $v = x_2 \bmod N$. If $v = r$, the verifier accepts the signature. Otherwise, the signature $S_j = (r, s_j)$ will be rejected

4.5. Correctness Analysis. If $S_j = (r, s_j)$ is a valid signature for the message m in the j-th period $(1 \leq j \leq T)$, then the hash value is $e'_j = H(j, m, r) = e_j$. Thus, $w_j = e'_j + s_j \bmod N = e_j + s_j \bmod N$, it results in the second equal sign in equation. Since the signature component $s_j = \mathrm{SK}_j k - e_j \bmod N$, it results in the third equal sign in equation. And because of the private key $\mathrm{SK}_j = a_0{}^{2^j} \bmod N$ and public key $\mathrm{PK} = a_0{}^{-1} G$, they result in the fifth and seventh equal signs in equation, respectively. In addition, we recall that the initial private key is $\mathrm{SK}_0 = a_0$ $(a_0 \in [1, n-1])$, where $n < N$, and we can get $a_0 < N$. It results in the sixth equal sign in equation.

$$\begin{aligned} X &= w_j{}^{2^{-j}}\mathrm{PK} = (e_j + s_j \bmod N)^{2^{-j}}\mathrm{PK} = (\mathrm{SK}_j k \bmod N)^{2^{-j}}\mathrm{PK} \\ &= (\mathrm{SK}_j \bmod N)^{2^{-j}}(k \bmod N)^{2^{-j}}\mathrm{PK} = \left(a_0{}^{2^j} \bmod N\right)^{2^{-j}} k^{2^{-j}}\mathrm{PK} \\ &= a_0 k^{2^{-j}}\mathrm{PK} = a_0 k^{2^{-j}} a_0{}^{-1} G = k^{2^{-j}} G. \end{aligned} \tag{8}$$

Thus, we have $(x_2, y_2) = X = k^{2^{-j}}G = (x_1, y_1)$. Since $r = x_1 \bmod N$ and $v = x_2 \bmod N$, we can get $v = r$. This means the verifier could believe and accept the signature $S_j = (r, s_j)$ of message m. Therefore, the construction of the signature is correct.

5. Proof of Security

5.1. Unforgeability

Theorem 1. *If the ECDLP problem is difficult, the improved scheme is existentially unforgeable under adaptive chosen-message attack in the ROM.*

Proof. Assume that adversary $\mathscr{A}$ breaks the improved scheme with advantage ε, the algorithm $\mathscr{B}$ simulates the challenger of $\mathscr{A}$ to perform the following operations:

(i) Setup phase: $\mathscr{B}$ sets the number of periods to be T and guesses the signature of the J-th $(0 \le J \le T)$ period forged by $\mathscr{A}$. $\mathscr{B}$ selects an elliptic curve E over the finite field $GF(q)$ and selects the base point G $(ord(G) = n)$ on the elliptic curve E. $\mathscr{B}$ selects the hash function $H(\cdot)$, and the initial private key a_0 calculates $a_0^{-1}G$. $\mathscr{B}$ sends the public key $a_0^{-1}G$ to $\mathscr{A}$

(ii) Query phase

(1) (Hash query): the hash query is firstly simulated, and $\mathscr{B}$ has a hash query list L_H, which is initially empty. In the j-th period, (m, e_j) is stored in the list L_H. When $\mathscr{A}$ performs a hash query of message m, the algorithm $\mathscr{B}$ firstly inspects whether m has appeared in the list L_H. If so, it returns e_j to $\mathscr{A}$ directly. Otherwise, $\mathscr{B}$ selects $e_j \longleftarrow \mathbb{Z}_n^* = \{1, 2, \cdots, n-1\}$ at random, stores (m, e_j) in the list L_H, and returns e_j to $\mathscr{A}$

(2) (Key leakage query): $\mathscr{A}$ submits breakin(j^+) $(1 \le j^+ \le T)$. If $j^+ \le J$, $\mathscr{B}$ reports failure and aborts. If $j^+ > J$, $\mathscr{B}$ calculates the key SK_{J+1}. If $j^+ \ne J+1$, $\mathscr{B}$ performs the KeyUpdate algorithm to generate SK_{j^+}, i.e., $\mathrm{SK}_{j^+} \longleftarrow \mathrm{Update}(\cdots \mathrm{Update}(SK_{J+1}))$. $\mathscr{B}$ returns SK_{j^+} to $\mathscr{A}$

(3) (Signature query): $\mathscr{A}$ submits (j, m) $(0 \le j \le j^+)$. $\mathscr{B}$ arbitrarily selects $k \in [1, n-1]$ and calculates $k^{2^{-j}}G = (x_1, y_1)$, $r = x_1 \bmod N$, $e_j = H(j, m, r)$, $s_j = \mathrm{SK}_j k - e_j \bmod N$. Then, $\mathscr{B}$ returns the signature (r, s_j) to $\mathscr{A}$

(iii) Forge phase: if the algorithm $\mathscr{B}$ does not abort in the query phase, the adversary $\mathscr{A}$ will output a period index $j^*(0 \le j^* < j^+)$, message m^*, and valid forgery (r^*, s_j^*) with an advantage of at least ε. If $j^* \ne J$, $\mathscr{B}$ reports failure and aborts; otherwise, $k = (e_j + s_j)a_0^{-2^j} \bmod N$. So, ECDLP problem can be solved

□

In the above simulation, if $\mathscr{B}$ guesses the period that $\mathscr{A}$ forge the signature, there is $j^+ > J$ in the key leaks query. That is, $\mathscr{B}$ can generate the private key SK_{j^+} without stopping and exiting. $\mathscr{B}$ has $1/T$ probability to guess correctly the period that $\mathscr{A}$ forge the signature. Therefore, $\varepsilon' = \varepsilon/T - \mathrm{negl}(n)$ is the minimum probability that the algorithm $\mathscr{B}$ successfully solves the ECDLP.

5.2. Forward Security

Theorem 2. *Because the quadratic congruence equation of modulo N is difficult, the improved scheme has forward security.*

Proof. If the adversary obtains the private key of the j-th period, he cannot use this private key to forge the signature of the $(j-1)$-th period. The reasons are as follows. □

Firstly, if the adversary wants to calculate the key SK_{j-1} of the $(j-1)$-th period by $\mathrm{SK}_j = \mathrm{SK}_{j-1}^2 \bmod N$, he must solve the quadratic congruence equation of modulo N. The difficulty of this problem is equivalent to the factorization problem.

Secondly, if the adversary wants to forge the signature (r, s_{j-1}) of the $(j-1)$-th period from $s_{j-1} = \mathrm{SK}_{j-1}k - e_{j-1} \bmod N$, he must know SK_{j-1}. According to the above, the adversary cannot find the private key SK_{j-1} of the $(j-1)$-th period by the private key SK_j of the j-th period.

We will prove this theorem as follows.

Assume that the adversary $\mathscr{O}$ attacks the forward security of the improved scheme, an algorithm $\mathscr{D}$ is constructed to solve the quadratic congruence equation by using $\mathscr{O}$ as a subroutine. $\mathscr{D}$ runs in two phases: the chosen-message-attack (CMA) and the forge. The algorithm has access to the signature oracle and the hash oracle. We make the public key PK $= v^{-2^{-(a+1)}}G$ be the input of the algorithm $\mathscr{D}$ in the CMA phase. Then, we begin running $\mathscr{O}$ in the CMA phase and return PK.

We choose a number $w \in Z_N^*$ randomly and let $v = w^2 \bmod N$. Then, we choose a value SK_j at random. We begin running $\mathscr{O}$ for the first time in the CMA phase and return PK. In the CMA phase, $\mathscr{O}$ can submit queries to the hash oracle and the signature oracle. Consequently, at the beginning of every period, $\mathscr{O}$ stays in the CMA phase, and the algorithm $\mathscr{D}$ is running. We randomly pick a value $\mathrm{SK}_j \in Z_N$. Here, we assume that the key will be revealed in both the previous and current periods if it is revealed at the beginning of the KeyUpdate.

(i) Query phase is as follows:

(1) (Signature query): we can simulate the signature oracle of $\mathscr{O}$ using our signature oracle easily and also simulate $\mathscr{O}$'s view of the signature algorithm. Assume that message m is queried to the signature oracle from $\mathscr{O}$, we query our signature oracle for the signature (r, s_j) of m and return the signature (r, s_j) to $\mathscr{O}$ as the answer of $\mathscr{O}$'s signature query

Then, we simulate $\mathscr{O}$'s view of the signature algorithm. At first, we simulate the generation of k and compute $k^{2-j}G = (x_1, y_1)$ to get r.

Table 1: Comparison of the computational complexity of the five schemes.

	Signature generation			Signature verification		
	DP	MM	MI	DP	MM	MI
Improved scheme	1	1	0	1	0	0
ECDSA scheme	1	2	1	2	2	1
Huiyan scheme	1	2	0	1	1	1
Keyuan scheme	4	3	1	0	3	0
Jinyuan scheme	2	3	0	2	1	1

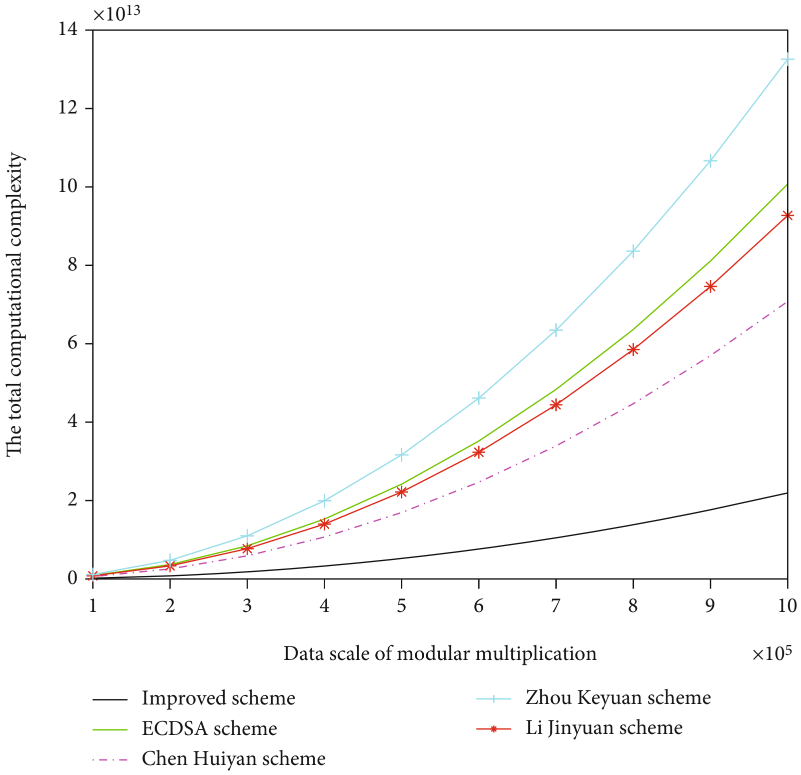

Figure 1: The total computational complexity of the five schemes.

(2) (Hash query): we can simulate the hash oracle of $\mathcal{O}$ using our hash oracle. Suppose (j, m, r) is queried to the hash oracle from $\mathcal{O}$, we query our hash oracle for the same input and return $\mathcal{O}$ the answer

(ii) Forge phase is as follows: suppose in the t-th period, we get into the breakin phase to know the current secret SK_t and return SK_t to $\mathcal{O}$. Suppose $\mathcal{O}$ outputs the forgery $(m, (r, s_t))$, we simply return $(m, (r, s_t))$ and take it as the output of $\mathscr{D}$

The success probability of algorithm $\mathscr{D}$ is almost the same as that of adversary $\mathcal{O}$. The only difference is as follows. In the simulation for the signature oracle above, all the values we used come from Z_N^*. However, in the real signature oracle, it is possible that some of the values it outputs come from Z_N not Z_N^*. Because the value k is randomly picked from Z_N in the signature phase, the probability that the value does not come from Z_N^* is negligible. If $N = pq$, the probability is at most $(p + q)/(pq) = (1/q) + (1/p)$. Therefore, if q_s is the total number of queries that submitted to the signature oracle, $q_s((1/q) + (1/p))$ is the success probability of the algorithm $\mathscr{D}$ exactly.

In summary, the improved scheme has forward security.

6. Efficiency Analysis

The wireless communication network is a restricted environment, i.e., time (key generation, signature and verification, etc.) and space (data memory, program memory, bandwidth, code and data length, etc.) limit the achievement of security goals. In addition, wireless devices have relatively low computing power and relatively small storage space. This requires that the digital signature scheme used in wireless communication network should be as efficient as possible while ensuring security. In this section, we analyze the efficiency of signature generation algorithm and signature verification algorithm.

We compare the calculation amount of the improved scheme with the ECDSA scheme, the Huiyan scheme [38], the Keyuan scheme [29], and the Jinyuan scheme [32]. The latter three also have forward security. The comparison results are shown in Table 1. In this table, "DP" represents the double-point arithmetic, "MM" represents the modular multiplication, and "MI" represents the modular inversion.

Assuming that the data scale of one modular multiplication is t, the complexity of one double-point arithmetic is $O(t^2)$, the complexity of one modular inversion is $O(9t^2)$, and the complexity of one modular multiplication is $O(t^2\log_2 t)$ [39]. According to Table 1, the total computational complexity of the improved scheme is $N_1 = O[(\log_2 n + 2)n^2]$, the total computational complexity of the ECDSA scheme is $N_2 = O[(4\log_2 n + 21)n^2]$, the total computational complexity of the Huiyan scheme is $N_3 = O[(3\log_2 n + 11)n^2]$, the total computational complexity of the Keyuan scheme is $N_4 = O[(6\log_2 n + 13)n^2]$, and the total computational complexity of the Jinyuan scheme is $N_5 = O[(4\log_2 n + 13)n^2]$. The total computational complexity of the five schemes is shown in Figure 1.

Modular multiplication and modular inversion are the main arithmetic that affects the total computational complexity. In the signature generation algorithm and signature verification algorithm, the improved scheme has one double-point arithmetic, three modular multiplications, and two modular inversions less than the ECDSA scheme. Besides, according to Figure 1, compared with the ECDSA scheme and other three schemes, the total computational complexity of the improved scheme is much smaller.

7. Conclusion

The potential security risks of wireless networks pose a serious threat to users' information security. Digital signature is the key technology for privacy protection in wireless communication network. However, the existing ECDSA scheme is computationally complex and does not have forward security. Aiming at the security problems of wireless communication network and the shortcomings of ECDSA scheme, we constructed an improved scheme and proved the security of the scheme. Compared with the ECDSA scheme and three existing forward secure schemes, our scheme not only meets the forward security but also reduces the requirement of computing power of the user in the wireless network. Therefore, our improved scheme is more suitable for wireless communication network. With its high security and high efficiency, our improved scheme can be widely applied to the scenarios involving privacy protection in wireless communication network. However, how to deploy our improved scheme to the application scenarios of wireless communication network will be our future work.

Data Availability

The calculation amount data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was sponsored by the National Natural Science Foundation of China (No. 12071112, No. 11471102, and No. 62102134), the Science and technology project of Henan Science and Technology Department (No. 222102210049), and the Science and Technology Project of Henan Educational Committee of China (No. 20A520012).

References

[1] L. Sun, Y. Qiandi, D. Peng, S. Subramani, and X. Wang, "Fogmed: a fog-based framework for disease prognosis based medical sensor data streams," *CMC-COMPUTERS MATERIALS & CONTINUA*, vol. 66, no. 1, pp. 603–619, 2021.

[2] Y. Wang, L. Sun, S. Subramani, and X. Wang, "Cab: classifying arrhythmias based on imbalanced sensor data," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 7, pp. 2304–2320, 2021.

[3] W. Tan, P. Huang, X. Li, G. Ren, Y. Chen, and J. Yang, "Analysis of segmentation of lung parenchyma based on deep learning methods," *Journal of X-ray science and technology*, vol. 29, no. 6, pp. 945–959, 2021.

[4] R. Zhonggang and Z. Donghai, "Selection of security elliptic curve in finite field gf(q)," *INFORMATION AND ELECTRONIC ENGINEERING*, vol. 7, no. 5, pp. 493–496, 2009.

[5] I. F. Blake, G. Seroussi, and N. P. Smart, *Elliptic curves in cryptography*, vol. 265, Cambridge university press, 1999.

[6] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[7] V. S. Miller, "Use of elliptic curves in cryptography," in *Lecture Notes in Computer Sciences; 218 on Advances in Cryptology*, 1986.

[8] Z. Keyuan, "Digital signature scheme based on elliptic curve and factoring," *Computer Science*, vol. 41, no. z1, pp. 366–368, 2014.

[9] N. Koblitz, A. Menezes, and S. A. Vanstone, "The state of elliptic curvecryptography," *Designs, Codes and Cryptography*, vol. 19, no. 2/3, pp. 173–193, 2000.

[10] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ecdsa)," *International Journal of Information Security*, vol. 1, no. 1, pp. 36–63, 2001.

[11] M. Bellare and S. K. Miner, "A forward-secure digital signature scheme," in *CRYPTO*, pp. 431–448, 1999.

[12] L. Mingxiang and A. Ni, "Construction of a lattice based forward-secure signature scheme," *Journal of Cryptologic Reseatch*, vol. 3, no. 3, pp. 249–257, 2016.

[13] Z. Ping and L. Yamin, "Forward secure elliptic curve digital signature scheme," *Computer Engineering and Applications*, vol. 1, no. 56, pp. 115–120, 2020.

[14] H. Krawczyk, "Simple forward-secure signatures from any signature scheme," in *CCS 2000, Proceedings of the 7th ACM Conference on Computer and Communications Security*, vol. 1-4pp. 108–115, Athens, Greece, 2000.

[15] M. Abdalla and L. Reyzin, "A new forward-secure digital signature scheme," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 116–129, Springer, Berlin, Heidelberg, 2000.

[16] T. Malkin, D. Micciancio, and S. M. More, "Composition and efficiency tradeoffs for forward-secure digital signatures," *IACR Cryptology ePrint Archive*, vol. 34, 2001.

[17] G. Itkis and L. Reyzin, "Forward-secure signatures with optimal signing and verifying," in *Advances in Cryptology - CRYPTO 2001, Proceedings*, p. 2139, 2001.

[18] A. Kozlov and L. Reyzin, "Forward-secure signatures with fast key update," in *Security in Communication Networks, Third International Conference, SCN 2003*, pp. 241–256, Amalfi, Italy, 2002.

[19] H. Fei, W. Chwan-Hwa John, and J. David Irwin, "A new forward secure signature scheme using bilinear maps," in *IACR Cryptology ePrint Archive*, p. 188, 2003.

[20] N. McCullagh and P. S. L. M. Barreto, "Efficient and forward-secure identity-based signcryption," in *IACR Cryptology ePrint Archive*, p. 117, 2004.

[21] B. G. Kang, J. H. Park, and S. G. Hahn, "A new forward secure signature scheme," in *IACR Cryptology ePrint Archive*, p. 183, 2004.

[22] X. Boyen, H. Shacham, E. Shen, and B. Waters, "Forward-secure signatures with untrusted update," in *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006*, pp. 191–200, Alexandria, VA, USA, 2006.

[23] J. A. Buchmann, E. Dahmen, and A. Hülsing, "XMSS - A practical forward secure signature scheme based on minimal security assumptions," in *Post-Quantum Cryptography -4th International Workshop, PQCrypto 2011. Proceedings*, pp. 117–129, 2011.

[24] J. K. Liu, T. H. Yuen, and J. Zhou, "Forward secure ring signature without random oracles," in *Information and Communications Security -13th International Conference, ICICS 2011, Proceedings*, pp. 1–14, 2011.

[25] Y. Yao-Chang, T.-Y. Huang, and T.-W. Hou, "Forward secure digital signature for electronic medical records," *Journal of Medical Systems*, vol. 36, no. 2, pp. 399–406, 2012.

[26] X. U. Guang-bao, J. I. A. N. G. Dong-huan, and L. I. A. N. G. Xiang-qian, "A strong forwardsecure digital signature scheme," *Computer Engineering*, vol. 39, no. 9, pp. 167–169, 2013.

[27] Y. Yao-Chang and T.-W. Hou, "An efficient forward-secure group certificate digital signature scheme to enhance EMR authentication process," *Medical & biological engineering & computing*, vol. 52, no. 5, pp. 449–457, 2014.

[28] Z. Ping, Z. Geng, M. Lequan, and L. Xiaodong, "An identity-based forward secure digital signature scheme," *Computer Applications and Software*, vol. 32, no. 11, pp. 289–292, 2015.

[29] Z. Keyuan, "A ttack analysis and improvement on a digital signature algorithm with message recovery," *Journal of Northwest Normal University*, vol. 52, no. 4, 2016.

[30] L. I. Yarong, L. I. Xiao, H. E. Mingxing, L. I. U. Xiaojian, and G. E. Lixia, "Security analysis and improvement of forward secure proxy signature scheme," *Computer Engineering and Applications*, vol. 52, no. 14, 2016.

[31] L. I. Shun-bo, H. U. A. N. G. Guang-qiu, and P. E. N. G. Jia-long, "Analysis and improvement for a digital signature scheme of forward security," *Computer Technology and Development*, vol. 26, no. 11, pp. 93–96, 2016.

[32] L. I. Jinyuan and M. I. A. O. Xianghua, "Analysis and improvement of forward-secure digital signature scheme," *Journal of Jilin University(Information Science Edition)*, vol. 35, no. 6, pp. 608–611, 2017.

[33] J. Kim and H. Oh, "Forward-secure digital signature schemes with optimal computation and storage of signers," in *ICT Systems Security and Privacy Protection -32nd IFIP TC 11 International Conference, SEC 2017, Proceedings*, pp. 523–537, 2017.

[34] Q. Xu, T. Chengxiang, F. Jun, F. Zhijie, and Z. Wenye, "Lattice-based forward secure and certificateless signature scheme," *Journal of Computer Research and Development*, vol. 54, no. 7, pp. 1510–1524, 2017.

[35] L. I. A. O. Xiaoping, "A forward-secure proxy blind signature scheme based on certificateless cryptosystem," *Modern Electronics Technique*, vol. 42, no. 1, pp. 91–94, 2019.

[36] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, CRC Press, 2007.

[37] W. Zhangyi, Y. Hantao, and Z. Huanguo, "Analysis of elliptic curve cryptosystem," *COMPUTER ENGINEER*, vol. 28, no. 5, pp. 161–163, 2002.

[38] C. Huiyan, Y. Yong, and W. Zongjie, "Liu Le, and Yang Yi. A forward-secure digital signature based on elliptic curve. Telecommunications," *Science*, vol. 31, no. 10, pp. 99–102, 2015.

[39] H. Yiliang, X. Yang, J. Hu, and P. Qingquan, "Improved elliptic curve digital signature algorithm," in *Computer Science*, vol. 382, pp. 377-378, China Computer federation, China Computer Federation, Changsha, 2003.