CONSTRUCTING DIGITAL SIGNATURE ALGORITHMS BASED ON NEW KEY SCHEMES

Nguyen Duc Thuy[1], *Bui The Truyen*[2], *Tong Minh Duc*[2], *Luu Hong Dung*[2]

Abstract

The paper proposes a method of constructing digital signature algorithms based on the new key schemes. The new key schemes are difficult problems that currently have no solution. That algorithm construction method with a new key scheme is to improve the security of digital signature algorithms. The new method is showed through the construction of two specific digital signature algorithms and the generation of highly secure signature algorithms that are constructed by this method is completely possible.

Index terms

Digital Signature Algorithm, Digital Signature Schema, Discrete Logarithm Problem, Root Problem.

1. Introduction

IN the [1], [3], [4], [5], the authors have proposed a method for constructing digital signature algorithms with the new key schemes. Analysis in [1], [3], [4], [5] has shown that the algorithms constructed by this new method have a highly secure efficiency against the private key attack and signature forgery attacks. However, the method presented in [1], [3], [4], [5] has an approach method from construct signature algorithm based on the root problem, so the generated algorithm has highly computational complexity and leads to low application efficiency. In these papers, the authors continue to propose a method of constructing digital signature algorithms based on the new key schemes, in terms of design principles, the proposed method has many similarities with the method shown in [1], [3], [4], [5] and the key schemes used here is based on a form of the difficult problem shown in [1], [3], [4], [5], but the approach of this method is from the construction of digital signature algorithm based on the discrete logarithm problem [2]. Therefore, constructed algorithms have a lower computational complexity so that they have a higher efficiency than those shown in [1], [3], [4], [5]. Improved implementation efficiency facilitates the algorithms constructed in this new method to approach practical applicability. Consequently, the proposed method can be considered as the perfection of the method presented in [1], [3], [4], [5] in the direction

[1]Ho Chi Minh City Technical and Economic College, [2]Le Quy Don Technical University

of improving the efficiency of application for digital signature algorithms constructed based on the new key schemes.

2. Construction of digital signature algorithms base on new key schemes

2.1. Proposed key schemes

2.1.1. 1 private key – 1 public key Schema (1-key scheme):

In this scheme, each signing object (signer) owns a pair of a private-public key. The private key of the signing object here is denoted as x and y for the public key, accordingly, where p is a large prime number (such that the discrete logarithm problem in Z_p is difficult to solve) and the value of x is an element of Z_p, then the value of y is calculated by the following formula:

$$y = x^x \bmod p.$$

It can be seen that, finding x with $y = x^x \bmod p$ is a new difficult problem, the existing algorithms for the root problem (RP) or discrete logarithms problem (DLP) cannot be applied to this problem. It also means that the problem used as the basis for the new 1-key scheme proposed here has a higher degree of difficulty than the known RP and DLP.

2.1.2. 2 private key – 2 public key Scheme (2-key scheme):

In this scheme, each signing object (signer) owns two pairs of the private-public key. These private key pairs are denoted as (x_1, x_2) and their values are elements of Z_p, where p is a large prime number. Then the accordingly public keys (y_1, y_2) is calculated according to the formula:

$$\begin{cases} y_1 = (x_1)^{x_1} \bmod p \\ y_2 = (x_1)^{(x_1)^{-1}.x_2} \bmod p \end{cases}$$

Similar to the 1-key scheme, it is easy to see that finding (x_1, x_2) from (p, y_1, y_2) is also a new difficult problem and have no solution currently and known algorithms for the problem with discrete logarithm or the root can not be applied to this problem.

2.2. Constructing the signature based on the 1-key scheme

The method of constructing a digital signature algorithm based on the 1-key scheme is presented through the construction of a specific signature algorithm, including:

2.2.1. Parameter and key generation algorithm:

In the method of constructing the new digital signature algorithm proposed here, the 1-key scheme is used to generate the private-public key pair of signing objects. The key generation algorithm is described in Table 1, where p and q are system parameters (domain parameters) generated by the digital authentication service provider (DASP), p is the prime number selected in a manner so that the DLP problem is difficult to solve

and $q|(q_1)$. x parameters is private keys and y is the public keys of each signing object in the system. To generate x, each signing object requires a number $\alpha \in Z_p^*$, x key is generated according to (1):

$$x = \alpha^{(p-1)/q} \bmod p \tag{1}$$

Then, the public key is generated from x, p according to (2):

$$y = x_x \bmod p \tag{2}$$

Table 1. Parameter and key generation algorithm

Input: lp, lq – length (in bits) of prime numbers p, q. **Output:** p, q, x, y.
[1]. **generate** p, q: len(p) = lp, len(q) = lq, $q\|(p-1)$ [2]. **select** α : 1 < α < p [3]. $x \leftarrow \alpha^{(p-1)/q} \bmod p$ [4]. **if** (x = 1) **then goto** [2] [5]. $y \leftarrow x^x \bmod p$
[6]. **return** $\{p, q, x, y\}$

<u>Note:</u> - len(.) : The function calculates the length (in bits) of an integer.

- p, q: System parameters/domain parameters.

- x: Private key.

- y: Public key of a signing object.

2.2.2. Signing algorithm:

Suppose that (R, S) is the signature on the message M of a signing object – who owns the key pair (x, y). The first component R is derived from a u value according to the formula:

$$R = x^u \bmod p \tag{3}$$

with u is a value in the range $(1, q)$. Similarly, the second component S is calculated from a v value according to the formula:

$$S \equiv x^v \bmod p \tag{4}$$

Here: v is also a chosen value in the range $(1, q)$.

Assume that the verification equation of the new digital signature algorithm has this form:

$S^E \equiv R^v \times (y)^{(R \times S mod p) mod q} \bmod p$ where E is the representative value of the message to be signed M, generated by the hash function H: $E = H(M)$

$$\text{Assume: } R \times S \bmod p = x^k \bmod p \tag{5}$$

in which, k is also a chosen value in the range $(1, q)$.

$$\text{Set} \qquad x^k \bmod p = T \tag{6}$$

Then, the verification equation can get into the form:

$$S^E \equiv R^y \times (y)^{(Tmodq)} \bmod p$$

From (1), (3), (4), (5) and (6) we have:

$$x^{v \times E} \equiv x^{u \times y} \times x^{x \times (Tmodq)} \bmod p$$

or:
$$x^{v \times Emodq} \equiv x^{u \times ymodq} \times x^{x \times Tmodq} \bmod p \quad (7)$$

From (7), we deduce:

$$v \times E \equiv (u \times y + x \times T) \bmod q \quad (8)$$

Besides, from (3), (4), (5) and (6) we have:

$$(v + u) \bmod q = k$$

Deduct:

$$u = (k - v) \bmod q \quad (9)$$

From (8) and (9) we have:

$$v \times E \bmod q = ((k - v) \times y + x \times T) \bmod q \quad (10)$$

From (10), deduce:

$$v = (k \times y + x \times T) \times (E + y)^{-1} \bmod q \quad (11)$$

From (9) and (11), the first component of the signature can be calculated according to (3): $R = x^{k-v} \bmod p$ and from (11), the second component can be calculated according to (4):

$$S = x^v \bmod p$$

From here, the signing algorithm is described in Table 2 as follows:

Table 2. Signing algorithm

Input: p, q, x, y, M. **Output:** (R, S).
[1]. $E \leftarrow H(M)$ [2]. **select** k: $1 < k < q$ [3]. $Z \leftarrow x^k \bmod p$ [4]. $v \leftarrow (k \times y + x \times T) \times (E + y)^{-1} \bmod q$ [5]. $u \leftarrow (k - v) \bmod q$ [6]. $R \leftarrow x^u \bmod p$ [7]. $S \leftarrow x^v \bmod p$
[8]. **return** (R, S)

Note:

- M: Message to sign, where: $M \in \{0, 1\}^\infty$.

- (R, S): signature on M.

2.2.3. Verification algorithm:

Because the verification equation of the new digital signature algorithm has the following form:

$$S^E \equiv R^y \times y^{(R \times S mod p) mod q} \text{ mod } p$$

Here, E is the representative value of the message to be validated: $E = H(M)$. If M and signature (R, S) satisfy the above equation, the signature is considered valid and the message will be validated to its origin and integrity. Otherwise, the signature is consider forged and the message is denied its origin and integrity. Therefore, if the left side of the equation is calculated according to:

$$A = S^E \text{ mod } p \quad (12)$$

and the right side is calculated according to:

$$B = R^y \times y^{\bar{T}} \text{ mod } p \quad (13)$$

whereas: $\bar{T} = (R \times S \text{ mod } p) \text{ mod } q$ (14)

The condition for a valid signature is A = B. Then, the verification algorithm of the new digital signature algorithm is described in Table 3 as follows:

Table 3. Verification algorithm

Input: p, y, M, (R,S). **Output:** *true / false*.
[1]. $E \leftarrow H(M)$ [2]. $A \leftarrow S^E$ mod p [3]. $\bar{T} \leftarrow (R \times S \text{ mod } p) \text{ mod } q$ [4]. $B \leftarrow R^y \times y^{\bar{T}}$ mod p
[5]. **if** (A = B) **then** {**return** *true*} **else** {**return** *false*}

Note:

- M, (R, S): message, signature to validate.

- If the result is true, the integrity and origin of M are confirmed. Otherwise, if the result is false, M is denied for its origin and integrity.

2.2.4. Correctness of the new algorithm:

What to solve here is: Let p, q are two prime numbers with: $q|(p-1)$, H: {0,1}$^* \mapsto Z_n, |q| \leq |n| < |p|, 1 < \alpha < p$, x = $\alpha^{(p-1)/q}$ mod p, $y = x^x$ mod p, E = $H(M)$, $1 < k < p, T = x^k$ mod p, $v = (k \times y + x \times T) \times (E + y)^{-1}$ mod q, $u = (k - v)$ mod q, $R = x^u$ mod p, $S = x^v$ mod p.

If $\bar{T} = (R \times S \text{ mod } p) \text{ mod } q$, $A = S^E$ mod p, $B = R^y \times y^{\bar{T}}$ mod p then A = B.

The correctness of the new algorithm is proven as follows: From (1), (2), (4), (11) and (12) we have:

$$A = S^E \bmod p = x^{v \times E} \bmod p = x^{(k \times y + x \times T) \times (E+y)^{-1} \times E mod q} \bmod p \quad (15)$$

with $T = x^k \bmod p$

Replace (3), (4), (5), (6), (9) and (11) into (14) we have:

$$\bar{T} = (R \times S \bmod p) \bmod q = (x^u \times x^v \bmod p) \bmod q$$

$$= (x^{(u+v) mod p} \bmod q) \bmod q = (x^k \bmod p) \bmod q = T \bmod q \quad (16)$$

Replace (1), (2), (3), (9) and (16) into (13) we have:

$$B = R^y \times y^{\bar{T}} \bmod p = x^{(k-v) \times y} \times x^{x \times (T mod q)} \bmod p$$

$$= x^{(k-v) \times y mod q} \times x^{x \times T mod q} \bmod p = x^{(k \times y + x \times T - v \times y) mod q} \bmod p$$

$$= x^{(k \times y + x \times T - y \times (k \times y + x \times T) \times (E+y)^{-1}) mod q} \bmod p$$

$$= x^{(k \times y + x \times T) \times (1 - y \times (E+y)^{-1}) mod q} \bmod p$$

$$= x^{(k \times y + x \times T) \times ((E+y) \times (E+y)^{-1} - y \times (E+y)^{-1}) mod q} \bmod p$$

$$= x^{(k \times y + x \times T) \times (E+y)^{-1} \times E mod q} \bmod p \quad (17)$$

From (15) and (17) deduce: $A = B$.

2.2.5. Security level of the new algorithm:

The security level of the new algorithm can be evaluated through its ability to defend against several types of attack such as:

- *Private key attack*

There are two types of private key attack:

+ *Attack on the key generation algorithm*: As discussed above, the key generation scheme here is a new form of difficult problem which has no solution currently.

+ *Attack on the signing algorithm*: The signing algorithm of the new algorithm shown that the private key of the signing object is used in steps [1], [2], [3], [5] of the algorithm. While in steps [1], [3], [5] the parameters Z, R and S are public, parameters k, u and v are private. Therefore, the difficulty of finding x from steps [1], [3], [5] of the signing algorithm is similar to finding x from the 1-key generation scheme. Also, in step [2] of the signing algorithm, v itself is a private parameter, so finding x from step [2] is impossible.

- *Signature forged attack*

The verification algorithm (Table 3) of the new algorithm shown a forged signature (R, S) will be recognized as a valid signature of an M message if it met the following condition:

$$S \equiv R \times y^{(R \times S mod p) mod q} \bmod p \quad (18)$$

From (18), if we choose R in advance and then calculate S, condition (18) will be:

$$S \equiv a^S \bmod p \quad (19)$$

Adversely, if we choose S in advance and then calculate R, condition (18) will be:

$$R \equiv b^R \bmod p \tag{20}$$

with a and b are constant, we can easily see that (19) and (20) are also difficult problems without any solution currently [1], [3], [4], [5].

2.3. Constructing digital signature algorithm based on 2-key scheme

The method of constructing the digital signature algorithm based on the 2-key scheme herein is also presented through several steps of constructing a digital signature scheme:

2.3.1. Parameter and key generation algorithm:

In this method of constructing digital signature algorithm, the 2-key scheme is used to generate pairs of the private and public keys of signing objects. The key generation algorithm is described in Table 4. Where, p and q are system parameters (domain parameters) generated by the DASP, p here is a prime number that needs to be chosen so that it is difficult to solve DLP and $q|(p-1)$. (x_1, x_2) pair is private keys and (y_1, y_2) are corresponding public keys of each signing object in the system. To generate x_1 key, each signing object requires a number $\alpha \in Z_p^*$, x_1 key is generated according to:

$$x_1 = \alpha^{(p-2)/q} \bmod p$$

x_2 key is a randomly chosen value in the range of $(1, q)$. Then public keys are generated from (x_1, x_2) according to (21) as follows:

$$\begin{cases} y_1 = (x_1)^{x_1} \bmod p \\ y_2 = (x_1)^{(x_1)^{-1} \times x_2 mod p} \bmod p \end{cases} \tag{21}$$

The parameter and key generation algorithm is likely to be described as shown in Table 4:

Table 4. Parameter and key generation algorithm

Input: p – prime number, lq – length (in bits) of prime number q. **Output:** q, x_1, x_2, y_1, y_2.
[1]. **generate** q: len(q) = lq, $q\|(p-1)$ [2]. **select** $\alpha : 1 < \alpha < p$ [3]. $x_1 \leftarrow \alpha^{(p-1)/q} \bmod p$ [4]. **if** $(x_1 = 1)$ **then goto** [2] [5]. **select** x_2: 1 < x_2 < q [6]. $y_1 \leftarrow (x_1)^{x1} \bmod p$, $y_2 \leftarrow (x_1)^{(x_1)^{-1} \times x_2 mod q} \bmod p$
[7]. **return** $\{q, x_1, x_2, y_1, y_2\}$

Note:

- len(.): The function calculates the length (in bits) of an integer.

- p, q: System parameters/domain parameters.

- x_1, x_2: Private key.

- y_1, y_2: Public key of a signing object.

2.3.2. Signing algorithm:

Suppose that (R, S) is the signature on the message M, u is a value in the range of $(1, q)$ and R is calculated from u according to:

$$R = (x_1)^u \bmod p \tag{22}$$

and S is calculated from v according to:

$$U = (x_1)^v \bmod p \tag{23}$$

Here: v is also a value in the range $(1, q)$.

Also, suppose that the verification equation of the algorithm has a form:

$$S^{y_1} \equiv R^{y_2} \times (y_1)^E \times (y_2)^{(R \times S mod p) mod q} \bmod p \text{ with } E = H(M)$$

and $$R \times S \bmod p = (x_1)^k \bmod p \tag{24}$$

in which $H(.)$ is the hash function, M is the message to sign and $k \in Z_q^*$.

Set $(x_1)^k \bmod p = T$ (25)

Then, the verification equation can be get into the form:

$$S^{y_1} \equiv R^{y_2} \times (y_1)^E \times (y_2)^{T mod q} \bmod p \tag{26}$$

From (21), (22), (23) and (26) we have:

$$(x_1)^{v \times y_1} \equiv (x_1)^{u \times y_2} \times (x_1)^{x_1 \times E} \times (x_1)^{((x_1)^{-1} \times x_2) \times (T mod q)} \bmod p$$

or:

$$(x_1)^{v \times y_1 mod q} \equiv (x_1)^{u \times y_2 mod q} \times (x_1)^{x_1 \times E mod q} \times (x_1)^{(x_1)^{-1} \times x_2 \times T mod q} \bmod p \tag{27}$$

From (27) we deduce:

$$v \times y_1 \equiv (u \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \bmod q \tag{28}$$

Besides, from (22), (23) and (24) we have:

$$(v + u) \bmod q = k$$

Deduct:

$$(k - v) \bmod q = u \tag{29}$$

From (28) and (29) we have:

$$v = (k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times (y_1 + y_2)^{-1} \bmod q \tag{30}$$

From (29) and (30), the first component of the signature can be calculated according to (22):

$$R = (x_1)^{k-v} \bmod p$$

and from (30) the second component can be calculated according to (23):

$$S = (x_1)^v \bmod p$$

From here, the signing algorithm is described in Table 5 as follows:

Table 5. Signing algorithm

Input: $p, q, x_1, x_2, y_1, y_2, M$. **Output:** (R,S).
[1]. $E \leftarrow H(M)$ [2]. **select** $k : 1 < k < q$ [3]. $T \leftarrow (x_1)^k \bmod p$ [4]. $v \leftarrow (k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times (y_1 + y_2)^{-1} \bmod q$ [5]. $u \leftarrow (k - v) \bmod q$ [6]. $R \leftarrow (x_1)^u \bmod p$ [7]. $S \leftarrow (x_1)^v \bmod p$
[8]. **return** (R,S)

Note: M: message to sign, with: $M \in \{0,1\}^{\infty}$; (R, S); signature of U on M.

2.3.3. Verification algorithm:

The verification algorithm of the new digital signature algorithm is supposed to be:

$$S^{y_1} \equiv R^{y_2} \times (y_1)^E \times (y_2)^{(R \times S mod p) mod q} \bmod p$$

Here, E is the representative value of the message to be validated: $E = H(M)$. If M and signature (R, S) satisfy the above equation, the signature is considered valid and the message will be validated to its origin and integrity. Otherwise, the signature is consider forged and the message is denied its origin and integrity. Therefore, if the left side of the verification equation is calculated according to:

$$A = S^{y_1} \bmod p \tag{31}$$

and the right side of the verification equation is calculated according to:

$$B = R^{y_2} \times (y_1)^E \times (y_2)^{\bar{T}} \bmod p \tag{32}$$

whereas $\bar{T} = (R \times S \bmod p) \bmod q$ (33)

The condition for a valid signature is: $A = B$.

Then, the verification algorithm of the new digital signature algorithm is described in Table 6 as follows:

Table 6. Verification algorithm

Input: $p, y_1, y_2,\ M,\ (R, S)$. **Output:** *true / false*.
[1]. $E \leftarrow H(M)$ [2]. $A \leftarrow S^{y_1} \bmod p$ [3]. $\bar{T} \leftarrow (R \times S) \bmod p) \bmod q$ [4]. $B \leftarrow R^{y_2} \times y_1^E \times (y_2)^{\bar{T}} \bmod p$
[5]. **if** A = B **then** {**return** *true*} **else** {**return** *false*}

Note:

- M, (R, S): message, signature to validate.

- If the result is true, the integrity and origin of M are confirmed. Otherwise, if the result is false M is denied for its origin and integrity.

2.3.4. Correctness of the new algorithm:

What to solve here is: Let p, q are two prime numbers with:

$q|(p-1), H : \{0,1\}^* \mapsto Z_n, |q| \leq |n| < |p|, 1 < \alpha < p, x_1 = \alpha^{(p-1)/q} \bmod p,$

$1 < x_2 < q, y_1 = x_1^{x_1} \bmod p, y_2 = (x_1)^{(x_1)^{-1} \times x_2 mod q} \bmod p, E = H(M),$

$1 < k < p$, $T = x_1^k \bmod p$, $v = (k \times y_2 + x_1 \times E + x_1^{-1} \times x_2 \times T) \times (y_1 + y_2)^{-1} \bmod q$,

$u = (k - v) \bmod q, R = x_1^u \bmod p, S = x_1^v \bmod p.$

If $\bar{T} = (R \times S \bmod p) \bmod q, A = S^{y_1} \bmod p, B = R^{y_2} \times y_1^E \times y_2^{\bar{T}} \bmod p$ then $A = B$.

The correctness of the new algorithm is proven as follows:

From (23), (30) and (31) we have:

$$A = S^{y_1} \bmod p = (x_1)^{v \times y_1} \bmod p = (x_1)^{v \times y_1 mod q} \bmod p$$
$$= (x_1)^{(k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times (y_1 + y_2)^{-1} \times y_1 mod q} \bmod p \quad (34)$$

From (22), (23), (25) and (33) we have:

$$\bar{T} = (R \times S \bmod p) \bmod q = ((x_1)^u \times (x_1)^v \bmod p) \bmod q$$
$$= ((x_1)^{(u+v) mod q} \bmod p) \bmod q = ((x_1)^k \bmod p) mod q = T \bmod q \quad (35)$$

Replace (21), (22), (30) and (35) into (32) we have:

$$B = R^{y_2} \times (y_1)^E \times (y_2)^{\bar{T}} \bmod p$$
$$= (x_1)^{u \times y_2} \times (x_1)^{x_1 \times E} \times (x_1)^{(x_1)^{-1} \times x_2 \times (T mod q)} \bmod p$$
$$= (x_1)^{u \times y_2 mod q} \times (x_1)^{x_1 \times E mod q} \times (x_1)^{(x_1)^{-1} \times x_2 \times T mod q} \bmod p$$
$$= (x_1)^{((k-v) \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) mod q} \bmod p$$
$$= (x_1)^{(k \times y_2 - v \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) mod q} \bmod p$$
$$= (x_1)^{(k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T - y_2 \times (k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times (y_1 + y_2)^{-1}) mod q} \bmod p$$
$$= (x_1)^{(k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times (1 - y_2 \times (y_1 + y_2)^{-1}) mod q} \bmod p$$
$$= (x_1)^{(k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times ((y_1 + y2) \times (y_1 + y_2)^{-1} - y_2 \times (y_1 + y_2)^{-1} mod q} \bmod p$$
$$= (x_1)^{(k \times y_2 + x_1 \times E + (x_1)^{-1} \times x_2 \times T) \times (y_1 + y_2)^{-1} \times y_1 mod q} \bmod p \quad (36)$$

From (34) and (36) deduce: $A = B$

2.3.5. Security level of the new algorithm:

The security level of the new algorithm can be evaluated through its ability to defend

against several types of attack such as:

- *Private key attack*:

There are two types of private key attacks: Attack on the key generation algorithm and attack on the signing algorithm, perform a similar analysis to the 1-key scheme, it shows that the private key attack against schemes constructed by this method is always challenged with a difficult problem without solution currently.

- *Attack on the forged signature*:

The verification algorithm (Table 6) of the new algorithm algorithm shows a forged signature (R, S) will be recognized as a valid signature of an M message if it met the following condition:

$$(S)^{y_1} \equiv (R)^{y_2} \times (y_1)^E \times (y_2)^{(R \times S mod p) mod q} \bmod p \tag{37}$$

From (37), if we choose R in advance and then calculate S, condition (37) will be:

$$(S)^{y_1} \equiv a^s \bmod p \tag{38}$$

Adversely, if we choose S in advance and then calculate R, condition (37) will be:

$$(R)^{y_2} \equiv b^R \bmod p \tag{39}$$

with a and b are constant, we can easily see that (38) and (39) are also difficult problems without any solution currently [1], [3], [4], [5].

2.4. Some evaluation of the application efficiency of signature schemes constructed based on the new method

The effectiveness of the digital signature algorithm can be evaluated through the cost of executing the signing algorithm, verification algorithm, and signature size that the schema generates. In this section, the effectiveness of the new algorithm will be evaluated and compared with the results in

2.4.1. Signature size:

It can be seen that with the same parameter set (p, q), the size of signatures generated by the 2 schemes herein and those in [1], [3], [4], [5] are equivalent.

2.4.2. Executing cost:

Executing cost or computation cost of the signing and verification algorithms can be evaluated by the number of computations to be performed or total computation time of the signing and verification algorithms, the convention of the use of symbols as bellows:

T_{exp}: The execution time of the modular exponentiation.

T_h: The execution time of the hash function.

T_{mul}: The execution time of the modulo multiplication.

T_{inv}: The execution time of the modular inverse.

Attention:

The key and parameter generation algorithm only needs to be executed once for all signing objects. Therefore, the computation cost for the key and parameter generation algorithm can be ignored when calculating the cost of performing the digital signature algorithm.

The executing cost for the signing algorithm (1) and verification algorithm (2) of algorithms in [1], [3], [4], [5] and the 2 new algorithms are shown in Table 7 as follows:

Table 7. Executing cost for digital signature algorithms

	T_{exp}		T_{mul}		T_{inv}		T_h	
	(1)	(2)	(1)	(2)	(1)	(2)	(1)	(2)
Algorithm [1]	8	3	7	2	3	0	1	1
Algorithm [2]	7	3	6	2	4	0	1	1
Algorithm [3]	6	3	5	2	5	0	1	1
Algorithm [4]	6	3	4	2	4	0	1	1
Algorithm with 1-key scheme	3	3	3	2	1	0	1	1
Algorithm with 2-key scheme	3	4	5	3	2	0	1	1

Note:

- The Algorithms [1], [3], [4] are signature algorithms proposed in [1], [3] and [4].

- The Algorithm [2] is a signature algorithm proposed in *"C. Constructing digital signature schema based on the difficulty of solving expanded root problem"* in [2].

- The Algorithm with 1-key and 2-key scheme is 2 signature algorithms constructed based on the newly proposed method here.

Remark:

Results in Table 3.1 shows that amount/duration to perform the calculations of the two algorithms constructed according to the new method is lower than those on the algorithms in [1], [3], [4], [5], thereby showing that the application efficiency of these algorithms is higher than those on the earlier proposed algorithms.

3. Conclusion

In this paper, the authors proposed a method to construct digital signature algorithms based on new key schemes to improve the security of the algorithms. The security of the algorithms constructed in this method is always guaranteed by difficult problems without a solution currently. Besides, the issue of improving efficiency for signature algorithms constructed by the proposed method in [1], [3], [4], [5] is also solved to meet the requirements of application algorithms in practice.

References

[1] Luu Hong Dung, Tong Minh Duc, and Luu Xuan Van. A new method for constructing digital signature schemes base on difficulty of the integer factorization and discrete logarithm root problems the zn. In *Fundamental and Applied IT Research Conference*, pages 1–9, 2018.

[2] Dung Luu Hong et al. A new construction method of digital signature algorithms. *International Journal of Computer Science and Network Security (IJCSNS)*, 16(12):53, 2016.

[3] Dung Luu Hong et al. A new digital signature scheme based on the hardness of some expanded root problems. *Procedia Computer Science*, 171:541–550, 2020.

[4] Luu Xuan Van and Luu Hong Dung. Constructing a digital signature algorithm based on the difficulty of some expanded root problems. In *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 190–195. IEEE, 2019.

[5] Luu Xuan Van, Luu Hong Dung, and Doan Van Hoa. Developing root problem aims to create a secure digital signature scheme in data transfer. In *2020 International Conference on Green and Human Information Technology (ICGHIT)*, pages 25–30. IEEE, 2020.

Manuscript received: 15-07-2020; Accepted: 30-10-2020

■

Nguyen Duc Thuy Graduated in Information Technology University of Foreign Languages; Informatics University of Ho Chi Minh City in 2005, Master degree from Academy of Economics and Business in 2013; Currently working in the Faculty of Information Technology - Ho Chi Minh City Technical and Economic College; Research field: information security.

Bui The Truyen graduated from Le Quy Don Technical University in 2000. He received a doctor's degree in analysis and information processing at Moscow Aviation Institute, Russia in 2008. Currently a lecturer at the Le Quy Don Technical University. His research interests are virtual reality simulation and information security. E-mail: truyenbuithe@lqdtu.edu.vn

Tong Minh Duc graduated from Le Quy Don Technical University in 2000. Received a doctorate from University of Electrical Engineering - Russia in 2007. Currently a lecturer at the Faculty of Information Technology - Le Quy Don Technical University. Research field: Image processing, object identification, information security safety. E-mail: ducmta@gmail.com

Luu Hong Dung graduated in Electronics and Communications from Le Quy Don Technical University in 1989, PhD at Le Quy Don Technical University in 2013; Currently working in the IT department - Le Quy Don Technical University; Research direction: Cryptography and information security. E-mail: luuhongdung@gmail.com

XÂY DỰNG THUẬT TOÁN KÝ SỐ DỰA TRÊN CÁC SƠ ĐỒ KHÓA MỚI

Tóm tắt

Bài báo đề xuất một phương pháp xây dựng các thuật toán chữ ký số dựa trên các sơ đồ khóa mới. Các sơ đồ khóa mới được sử dụng ở đây thực chất là một dạng bài toán khó mà hiện tại còn chưa có cách giải. Phương pháp xây dựng thuật toán với các sơ đồ tạo khóa mới như thế được đề xuất nhằm mục đích nâng cao mức độ an toàn cho các thuật toán chữ ký số. Phương pháp mới đề xuất ở đây được trình bày thông qua việc xây dựng 2 thuật toán chữ ký số cụ thể, song hoàn toàn có thể tạo ra một lớp thuật toán chữ ký có độ an toàn cao cũng bằng chính phương pháp này.