A new construction method of the digital signature scheme based on the discrete logarithm combining find root problem on the finite field F_p

Nguyen Kim Tuan[1*], Nguyen Vinh Thai[2], Luu Hong Dung[3]

[1]Duy Tan University;
[2]Academy of Military Science and Technology;
[3]Military Technical Academy.
[*]Corresponding author: nguyenkimtuan@duytan.edu.vn
Received 30 Aug 2022; Revised 10 Nov 2022; Accepted 28 Nov 2022; Published 20 Dec 2022.
DOI: https://doi.org/10.54939/1859-1043.j.mst.FEE.2022.164-170

ABSTRACT

The article proposes a method to build a signature scheme based on a new hard problem, called the logarithm problem with roots on the finite field F_p. Now, this is a hard problem belonging to the class of unsolvable problems, except for the "brute force" method. Therefore, building a digital signature scheme based on the difficulty of this problem will most likely allow for improving the security of the digital signature algorithm according to the proposed new method. In addition, the method of building signature schema here can be applied to develop a class of signature algorithms suitable for applications with high requirements for security in practice applications.

Keywords: Discrete logarithm problem (DLP); Digital signature algorithm; Digital signature schemes; Asymmetric - key cryptosystems.

1. INTRODUCTION

Improving the security of the digital signature scheme is always a critical issue when the ability to attack public key cryptosystems in general and digital signature systems, in particular, is continuously increasing thanks to advancements in science and technology. The published research results [1-8] show that the basic approach to improving the security of signature schemes is mainly based on the difficulty of solving 2 problems simultaneously in mathematics. This primarily focuses on two problems: the problem of analyzing a large integer into prime factors and the problem of discrete logarithms on the prime finite field F_p. However, once an attacker is competent enough to solve one problem, it will, in principle, also solve the other, so such an approach makes no practical sense.

In this article, the authors propose a method to build a digital signature scheme based on a new type of hard problem that currently has no solution. As a result, the proposed new solution-built scheme is resistant to known secret key attacks and signature forgery attacks in real applications.

2. DISCRETE LOGARITHM PROBLEM COMBINED WITH FINDING ROOT ON FINITE FIELD F_P - A NEW TYPE OF HARD PROBLEMS

The hard problem as a basis for building a signature scheme here is called a discrete logarithm problem combined with finding root on finite field F_p [9]. This problem is formed based on a discrete logarithm problem of the form:

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

where p is a prime number, g is the generator of F_p, and x is the value found from the public parameters g, p, y.

From the discrete logarithm problem on F_p, we see that if the parameter g is also kept secret, the logarithm problem on F_p will become an unsolvable problem. In the simplest case, we choose the secret key x itself for the role of parameter g. Then the problem can be stated as follows: let p be a prime number, and y belongs to F_p, find x satisfying the following equation:

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

It can also be derived from the root problem: find the value of x that satisfies the equation:

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

where p is a prime number and τ is a value in the range $(1, p-1)$. We also get the same result as above if the parameter τ is kept secret. In the simplest case, it is possible to choose the secret parameter x for the role of τ. Then, the problem of taking root on F_p also becomes an unsolvable problem of the form:

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

With the above approach, this problem is called a discrete logarithm problem combined with finding root on finite field F_p, or in short, a logarithm problem with roots.

This new hard problem can be stated in the first form as follows:

Form 1: Given a prime number p and a positive integer y in F_p, find the number x that satisfies the following equation: $y = x^x \bmod p$.

Another approach also derived from the above two problems is:

If the left side of equality: $y = g^x \bmod p$ in the discrete logarithm problem is a variable of the form: $x^b \bmod p$, then the logarithm problem becomes unsolvable, and then this problem has the form: $x^b \bmod p = g^x \bmod p$.

Similarly, if the left side of the equality: $y = x^\tau \bmod p$ in the finding root problem is a variable of type: $a^x \bmod p$, then the finding root problem also becomes an unsolvable problem, get: $a^x \bmod p = x^\tau \bmod p$.

With this approach, we can state the second form of the new hard problem as follows:

Form 2: Given p is a prime number, a and b are numbered in F_p, find the number x satisfying the following equation: $a^x \equiv x^b \bmod p$.

Currently, algorithms for discrete logarithm problems or rooting on F_p do not apply to this problem. That is, there is no solution to this problem other than the "brute force" method with computational complexity $\boldsymbol{O}(2^n)$, here: $n = |p|$.

3. CONSTRUCTION METHOD OF DIGITAL SIGNATURE SCHEME BASED ON THE DISCRETE LOGARITHM COMBINING, FIND ROOT PROBLEM

The method of construction a digital signature scheme proposed here is presented by building a signature scheme based on the difficulty of the logarithm problem with roots on F_p. *Form 1* is used to form the private and public key pairs of the signing objects in the key generation algorithm, the signature components are also generated by the signing algorithm from *Form 1*. *Form 2* is used as the basis to build the algorithm to verify the signature of the scheme.

The new signature scheme proposed here includes the parameter and key generation algorithms, the signing algorithm, and the signature verifying algorithm built as follows:

3.1. Domain parameter and key generation algorithm

The primes p and q as system or domain parameters are chosen similarly to the US DSS [10] standard or the Russian Federation GOST R34-90.10 [11]. To generate a private/public key pair, each signer must choose a value first and then compute the secret key x by $x = \alpha^{\frac{p-1}{q}} \bmod p$.

The public key y is generated from x and p by:

$$y = x^x \bmod p \quad (1)$$

Then the algorithm for generating parameters and keys is described as follows:

Algorithm 1:

input: lp, lq

output: x, y.

Step 1. Choose prime divisor q, where: $2^{159} < q < 2^{160}$

Step 2. Choose integer t, where $0 \le t \le 8$. And prime number p, where:

$$2^{511+64t} < p < 2^{512+64t} \text{ so that } q|(p-1)$$

Step 3. Select α: $1 < \alpha < q$

Step 4. Compute: $x = \alpha^{\frac{p-1}{q}} \bmod p$. If $x = 1$ or $x = q$ then goto Step 3

Step 5. Compute: $y = x^x \bmod p$. If $y = 1$ then choose goto Step 3

Step 6. Select hash function: $H: \{0,1\}^* \mapsto Z_h, q < h < p$

Step 7. Return $\{p, q, x, y, H\}$

Note:

- $len(.)$: function to calculate length (in bits) of an integer;
- l_p, l_q: length (in bits) of prime numbers p and q;
- p, q: system parameter/domain parameter;
- x, y: private and public key of the signer.

3.2. Signing

Assuming (r, s, z) is the signature on the message to be signed M and the condition for (r, s, z) to be recognized as valid is:

$$(z)^s \equiv (r)^e \times (y)^{(r \times z^s) mod\ p} \bmod p \quad (2)$$

Here, e is the representative value of the message to be signed M (the hash value of M). The z component of the signature is computed according to the following formula:

$$z = x^k \bmod p \quad (3)$$

where k is a randomly chosen value in the range $(1, q)$.

Also, assume that the component r is generated from a value u according to the formula:

$$r = z^u \bmod p \quad (4)$$

Here, the u is also randomly chosen in the range $(1, q)$.

The generation of the u component of the signature is done as follows:

From (4), we have:

$$r \times z^s \bmod p = z^{u+s} \bmod p \quad (5)$$

Set:

$$v = (u + s) \bmod q \quad (6)$$

Then (5) will become:

$$r \times z^s \bmod p = z^v \bmod p \quad (7)$$

From (1), (2), (3), (4) and (7) we have:

$$x^{k \times s} \equiv x^{k \times u \times e} \times x^{x \times (z^v \bmod p)} \bmod p \quad (8)$$

From (8) we deduce:

$$k \times s \equiv \left(k \times u \times e + x \times (z^v \bmod p)\right) \bmod q \quad (9)$$

On the other hand, from (6) we have:

$$u = (v - s)\ mod\ q \quad (10)$$

Substituting (10) into (9) we get:

$$k \times s \equiv \left(k \times (v - s) \times e + x \times (z^v\ mod\ p)\right)\ mod\ q \quad (11)$$

From (11) deduce:

$$s = \left(k \times v \times e + x \times (z^v\ mod\ p)\right) \times \left(k \times (e + 1)\right)^{-1} mod\ q \quad (12)$$

From (10) and (12), the r value is calculated according to:

$$r = z^{(v-s)}\ mod\ p \quad (13)$$

Then, the signing algorithm is described as follows:

Algorithm 2:

input: p, q, x, y, M

output: (r, s, z).

Step 1. Compute: $e = H(M)$

Step 2. Choose a random integer k, v in the interval $(1, q)$.

Step 3. Compute: $z = x^k\ mod\ p$

Step 4. Compute: $s = \left(k \times v \times e + x \times (z^v\ mod\ p)\right) \times \left(k \times (e + 1)\right)^{-1} mod\ q$

Step 5. Compute: $r = z^{(v-s)}\ mod\ p$

Step 6. Return (r, s, z)

Note:

- M: message to sign, with $M \in \{0,1\}^{\infty}$;
- (r, s, z): signature on the message to be signed M.

3.3. Verifying

The verification algorithm of the schema is constructed on the assumption:

$$(z)^s \equiv (r)^e \times (y)^{(r \times z^s) mod\ p}\ mod\ p \quad (14)$$

That is, if M and the signature (r, s, z) satisfy the equality (14), then the signature is considered valid, and the message M is verified for origin and integrity. Otherwise, the signature is considered forged, and the message to be verified is denied in terms of origin and integrity. Therefore, if the left-hand side of the verification equality is computed as:

$$A = z^s\ mod\ p \quad (15)$$

And the right-hand side of the verification equality is:

$$B = r^e \times y^{(r \times z^s)\ mod\ p}\ mod\ p \quad (16)$$

Then the condition for a valid signature is: A = B

The verifying algorithm of the scheme will then be described as follows:

Algorithm 3:

input: $p, q, y, M, (r, s, z)$.

output: $True/False$.

Step 1. Compute: $e = H(M)$

Step 2. Compute: $A = z^s\ mod\ p$

Step 3. Compute: $B = r^e \times y^{(r \times z^s)\ mod\ p}\ mod\ p$

Step 4. If $A = B$ then return $True$ else return $False$

3.4. The correctness of the proposed new signature scheme construction method

What needs to be proved here is:

If $A = z^s\ mod\ p$ and $B = r^e \times y^{(r \times z^s)\ mod\ p}\ mod\ p$ then: A = B.

Substituting (3) into (15), we have:

$$A = z^s \bmod p = x^{k \times s} \bmod p$$

Similarly, by substituting (1), (3), (4), (7), and (10) into (16), we get:

$$B = r^e \times y^{(r \times z^s) \bmod p} \bmod p = x^{k \times (v-s) \times e + x \times (z^v \bmod p)} \bmod p$$

Now what to prove would be:

$$k \times s \equiv \big(k \times (v - s) \times e + x \times (z^v \bmod p)\big) \bmod q$$

It is equivalent to:

$$s \times k \times (e + 1) \equiv \big(k \times v \times e + x \times (z^v \bmod p)\big) \bmod q$$

Therefore, it can be re-stated what needs to be proved as follows:

If

$$C = s \times \big(k \times (e + 1)\big) \bmod q \tag{17}$$

and

$$D = \big(k \times v \times e + x \times (z^v \bmod p)\big) \bmod q \tag{18}$$

then: $C = D$.

Indeed, substituting (12) into (17) we get:

$$\begin{aligned} C &= s \times \big(k \times (e + 1)\big) \bmod q \\ &= \big(k \times v \times e + x \times (z^v \bmod p)\big) \times (k \times (e + 1)^{-1}) \times \big(k \times (e + 1)\big) \bmod q \\ &= \big(k \times v \times e + x \times (z^v \bmod p)\big) \bmod q \end{aligned} \tag{19}$$

From (18) and (19) deduce: $C = D$.

Thus, the correctness of the schema has been proved.

3.5. The security level of the New Scheme

The security of a digital signature scheme can be assessed on several bases as follows:

a) Against to secret key attack

A secret key attack can be performed on the key generation algorithm (**Algorithm 1**) and Step 3, and Step 4 of the signing algorithm (**Algorithm 2**). In Step 3, since k is also a secret parameter, finding x from Step 3 of the Signing algorithm is as difficult as finding x from the Key generation algorithm, as it is known this is a type of hard problem that currently there is no solution. In Step 4 of the Signing algorithm, in addition to x being the secret parameter to be found, k and v are also secret parameters, even if v is found from Step 5 by solving the DLP, then finding x from Step 4 of the Signing algorithm is also impossible. Thus, to find the secret key, the attacker is forced to solve the above hard problem by the "brute force attack" method with a computational complexity of about $\boldsymbol{O}(2^n)$, with $n = |p|$.

b) Signature forgery attack

From the verifying algorithm (**Algorithm 3**) of the proposed scheme, a set of 3 values (r, s, z) will be recognized as a valid signature with the message to be verified M if the condition is satisfied:

$$z^s \equiv r^e \times y^{(r \times z^s) \bmod p} \bmod p \tag{20}$$

From (20) shows pre-selecting 2 out of 3 values (r, s, z) and then calculating the remaining 3rd value is the 2nd form of the hard problem mentioned in Section 2, as it is known this is a type of hard problem that currently in mathematics there is no other solution, than the "brute force attack" method.

Thus, to generate a forged signature corresponding to a given message, the attacker has no choice but to randomly choose a set of three values (r, s, z) satisfying (20), which in fact, this is also a "brute force attack" method.

3.6. The performance of the algorithm

The effectiveness of the proposed scheme is evaluated by comparing the implementation cost of this scheme with the implementation cost of the DSA [10] and GOST R34-10.94 [11] digital signature scheme.

The computational cost (or cost) is the number of operations to be performed, where the symbols are defined as follows:

N_{exp}: the number of modulo exponentiations.

N_h: the number of hash operations.

N_{mul}: the number of modulo multiplications.

N_{inv}: the number of modulo division (inversion).

Note:

The algorithm for generating parameters and keys only needs to be done once for every schema. Therefore, the computational cost for the key and parameter generation algorithms can be ignored when comparing the costs of the schemas.

The cost for the signing algorithm and the verification algorithm of the DSA and GOST R34.10-94 compared with the proposed scheme (MTA V22.09-11) is shown in table 1 and table 2 as follows:

Table 1. *Cost of signature schemes.*

	N_{exp}	N_{mul}	N_{inv}	N_h
DSA	1	2	1	1
GOST R34.10 - 94	1	2	0	1
MTA V22.09 - 11	3	5	1	1

Table 2. *Cost of verifying schemes.*

	N_{exp}	N_{mul}	N_{inv}	N_h
DSA	2	3	1	1
GOST R34.10 - 94	3	3	0	1
MTA V22.09 - 11	3	2	0	1

Comment:

Comparing the cost of the proposed scheme (MTA V22.09-11) with the DSA and GOST R34.10-94, as shown in table 1 and table 2, it shows that the performance of the proposed scheme is lower than that of DSA and GOST R34.10-94. It can be seen that this is the cost of improving the security of the proposed scheme.

4. CONCLUSIONS

In this paper, the authors propose a method to construct a new digital signature scheme based on a new type of hard problem (discrete logarithm problem combined with finding root on finite field F_p) to improve security for the digital signature scheme. Now, this is a type of hard problem that belongs to the class of unsolvable problems. On the other hand, the signature scheme construction here is done according to a completely new method. It is an essential factor that allows for improving the security of the digital signature scheme according to this new method. From the proposed new method, it is possible to deploy a family of highly secure digital signature schemes suitable for different options in practical applications.

REFERENCES

[1]. W. Diffie & M. Hellman, *"New Directions in Cryptography"*, IEEE Trans. On Info. Theory, IT-22(6):644-654, (1976).

[2]. T. ElGamal, *"A public key cryptosystem and a signature scheme based on discrete logarithms*", IEEE Transactions on Information Theory. Vol. IT-31, No. 4. pp.469-472, (1985).

[3]. Mark Stamp, Richard M. Low, "*Applicd cryptanalysis: Breaking Ciphers in the Real World*", John Wiley & Sons, Inc., ISBN 978-0-470-1.

[4]. B. Arazi, *"Integrating a key distribution procedure into the digital signature standard*", *Electronics Letters*, Vol. 29(11), pp.966-967, (1993).

[5]. Do Viet Binh, *"Authenticated key exchange protocol based on two hard problems*", Tạp chí nghiên cứu khoa học và công nghệ quân sự, số 50, trang 147-152, (2017).

[6]. Đỗ Việt Bình, Nguyễn Hiếu Minh, *"Phát triển giao thức trao đổi khóa an toàn dựa trên 2 bài toán khó*", Tạp chí Nghiên cứu KH&CN quân sự, Số Đặc san CNTT, (2018) (in Vietnamese).

[7]. Nguyễn Vĩnh Thái, Lưu Hồng Dũng, *"Xây dựng giao thức trao đổi khóa an toàn dựa trên tính khó của việc giải đồng thời hai bài toán logarit rời rạc và phân tích số/khai căn cho các hệ mật khóa đối xứng*", Tạp chí Nghiên cứu KH&CN quân sự, Số Đặc san CNTT, (2019) (in Vietnamese).

[8]. *"Cryptography and Network Security: Principles and Practice"*, 7th Edition, ISBN 978-0-13-444428-4, by William Stallings 2017.

[9]. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Ar2.pdf.

[10]. National Institute of Standards and Technology, FIPS PUB 186-4, 2013.

[11]. GOST R 34.10-94, Russian Federation Standard Information Technology. Cryptographic Data Security, Produce and Check Procedures of Electronic Digital Signature based on Asymmetric Cryptographic Algorithm, Government Committee of Russia for Standards, (1994) (in Russian).

TÓM TẮT

Phương pháp xây dựng lược đồ chữ ký số mới dựa trên bài toán logarit kết hợp khai căn trên F_p

Bài báo đề xuất một phương pháp xây dựng lược đồ chữ ký dựa trên một bài toán khó mới, ở đây gọi là bài toán logarit kết hợp khai căn trên trường hữu hạn F_p. Hiện tại, đây là bài toán khó thuộc lớp bài toán không giải được, ngoại trừ phương pháp "vét cạn". Do đó, việc xây dựng lược đồ chữ ký số dựa trên tính khó của bài toán này nhiều khả năng sẽ cho phép nâng cao độ an toàn của thuật toán chữ ký số theo phương pháp mới đề xuất. Ngoài ra, phương pháp xây dựng lược đồ chữ ký ở đây có thể áp dụng để phát triển một lớp thuật toán chữ ký phù hợp với các ứng dụng yêu cầu cao về độ an toàn trong thực tế.

Từ khóa: Discrete logarithm problem (DLP); Digital signature algorithm; Digital signature schemes; Asymmetric - key cryptosystems.