A Development of DSA Digital Signature Scheme Based on Ring Z_n

Le Van Tuan[1*], Ta Minh Thanh[2], Leu Duc Tan[3]

[1] *Military science Academy, No. 322, Le Trong Tan, Hoang Mai, Hanoi, Viet Nam*

[2] *Le Qui Don University, No. 236, Hoang Quoc Viet, Bắc Từ Liêm, Hanoi, Viet Nam*

[3] *Institute of cryptographic Technology, 141 Chien Thang, Thanh Tri, Ha Noi, Viet Nam*

Received: August 11, 2018; Accepted: November 26, 2018

Abstract

We have known, the DSA scheme is not secure in situations of coinciding or revealing of session key. In this paper, we propose a solution that improves the DSA digital signature scheme on ring Z_n The idea of our solution is developing the DSA scheme, in which its security is based on discrete logarithm problem on ring Z_n(DLP_n), with the modulo number n is a product of two distinct primes. The proposed scheme is secure from the situations of revealing or coinciding of session key, for this advantage, it can be applied into practice.

Keywords: Digital Signature Scheme, Discrete logarithm problem, Hash Function.

1. Introduction

Nowadays, the digital signature has played an important role for authentication; therefore, it has been being applied in many organizations and countries in the world. Since Elgamal proposed a digital signature scheme in 1985 [1-2], until now there have been many its variants that have been proposed by the scientists, such as: the Schnorr signature scheme in 1990 [3-5], the DSA signature scheme in 1994[6]. In general, all digital signature schemes are based on the discrete logarithm on field Z_p (p is a prime number). Furthermore, in signature schemes on field Z_p, the order of primitive element (denoted by g) can't be kept secret, that lead to the schemes on field Z_p are insecure from revealing or coinciding session key. Recently, there have been many research results against these types of attacks [7-13]. Such as, in [10], In order to resolve the security decline caused by the ElGamal signature scheme which uses only one random number, a modified scheme was proposed by Li Xiao-fei, Shen Xuan-jing and Chen Hai-peng. Their proposed scheme should be avoided to use the one k-value in order to sign more than one massages, however, in their scheme, if hackers figure out the value of the private key x then they can forge any a signature of the modified scheme. In [13] pointed out the DSA scheme is insecure from "Research on L3 Cache Timing Attack". The general characteristics of some schemes on field Z_p are public the order of primitive element g, that lead to the insecurity when the session key is revealed or coincided. In order to deal with insecure situations caused by the revealing or coinciding of session keys, recently, scientists have developed signature schemes and some other security systems on the ring Z_n [14-19]. Our proposed scheme's the security is based on discrete logarithm problem on ring Z_n, in which, the module number n is a product of two prime. The proposed scheme is secure from attacking basing revealing or coinciding of session key. Furthermore, in our scheme, if the secret key is revealed, it can't be forged by adversaries. Some important contributions of this paper are as follow:

The first, our proposed scheme is taken full advantage of specific characteristics of the DSA scheme such as the time complexity. Furthermore, it overcomes disadvantage of the DSA scheme such as our scheme is secure from attacks based on revealing key and coinciding of session key, even when the secret key is revealed, adversaries is still difficult to forge signature.

The second, in our signature scheme, the inverse element of the secret key (denoted by x^{-1}mod t) is pre-calculated, so the signature generation in our scheme is faster than the signature generation in the DSA scheme and Elgamal scheme and the DSA scheme.

The third, in term of time complexity, our scheme is similar to the DSA scheme (Table 1, Fig 2, fig 3). In term of memory space complexity, our scheme is more complex than the DSA scheme. Because each member of the system that has to use a separate module number, which prevent attacks from using the common module number

Finally, in our scheme, the first component of the signature is hashed together with the message; the output result is used for calculating the second

* Corresponding author: Tel: (+84) 989394556
Email: levantuan71@yahoo.com

component which can make the link between the first component and the second component more complicated.

The rest of paper is organized as follows: In section 2, we give some related works. The section 3, we present the proposed scheme. Finally, we present some test results, conclusion and future works.

2. Related works

2.1. Some definitions

In this section, we are going to define some functions which are used for following sections. In addition, we present the DSA scheme because our proposed scheme will be compared with them.

Definition 1. Function Num: $\{0,1\}^{\infty} \rightarrow Z$ $\mathrm{Num}(b_k b_{k-1} \dots b_0) = a$, in which, number a is calculated by formula follow: $a = b_0 + 2^1 b_1 + \dots + 2^k b_k$

Definition 2. Function str(a): $Z_{\geq 0} \rightarrow \{0,1\}^{\infty}$ it is a function that returns a binary number corresponding to a non-negative integer.

Definition 3. Random (a,b): Assume that a, b are positive integers. Random(a,b) is a function that returns an positive integer in (a, b).

Definition 4. Len(t): The len(t) is a function that returns the value m's number of bit that is in binary form.

Definition 5. A||B is the concatenation operation the string A with the string B.

2.2. The DSA scheme

The parameters of the DSA [6] signature scheme is as follow:

Let p is a odd prime, $\mathrm{len}(p) = L$.

Let q is a prime factor of $p - 1$, $\mathrm{len}(q) = N$.

g is a primitive element of subgroup, denoted by Z_q on finite field Z_p, $0 < g < p$,

Signer's secret key is x, x is chosen randomly in $(1, q-1)$.

Signer's public key is (p, q, g, y), where $y = g^x \bmod p$.

k is a session key for each message which is chosen randomly or pseudo randomly in (1, q–1).

Let H: $\{0,1\}* \rightarrow \{0,1\}^l$ be a hash function, in practice $l = 160$.

Algorithm 1: Generation signature

Suppose T stands for the message to be signed, the generation a signature for a messageT, T∈ $\{0,1\}^*$is as follow:

Input: T.

Output: (r, s) is the signature of T.

1. $z \leftarrow \mathrm{Num}(H(T))$.
2. $k \leftarrow \mathrm{Random}(1, q)$.
3. $r \leftarrow (g^k \bmod p) \bmod q$.
4. $w \leftarrow (z + x.r) \bmod q$.
5. if $(r = 0)$ or $(w = 0)$ then goto 2.
6. $s \leftarrow k^{-1}.w \bmod q$.
7. Return (r, s).

Algorithm 2: Signature verification

Input: pair (r, s) is the signature of T.

Output: "accept" or "reject".

1. $w \leftarrow s^{-1} \bmod q$.
2. $z \leftarrow \mathrm{Num}(H(T))$.
3. $u_1 \leftarrow (z.w) \bmod q$.
4. $u_2 \leftarrow (r.w) \bmod q$.
5. $v \leftarrow (g^{u_1}.y^{u_2} \bmod p) \bmod q$.
6. if $(v = r)$ then return "accept" else return "reject".

Security analysis:

The security of the DSA signature scheme is based on the hardness of the DL problem in finite prime field. In the DSA scheme, the order of the primitive element g is public that makes the DSA signature scheme insecure in some of the situations as following:

The first situation: The session key k is revealed, the secret key x is calculated by the following formula:

$s = (k^{-1}(z + r.x)) \bmod q$, the secret key x is calculated easily using the following formula:

$$x = (s.k - z).r^{-1} \bmod q \quad (1)$$

The second situation: Using the same the session key k (the session key k is coinciding). Suppose that the session key k ($k \in Z_q$) is same in two signatures. Then the first component of the signature r is culculated by $r = g^k \bmod q$ it is same in two signatures, but the second components are different. Basing on the first component r, the second component is calculated by the following formula:

$$s = k^{-1}(z + r.x) \bmod q$$

$$\leftrightarrow k = s^{-1}(z + r.x) \bmod q$$

$$s' = k^{-1}(z' + r.x) \bmod q$$

$$\leftrightarrow k = (s')^{-1}(z' + r.x) \bmod q$$

Basing on value of k, the equation is established as below:

$$s^{-1}(z + r.x) = (s')^{-1}(z' + r.x) \bmod q$$

$$\leftrightarrow s^{-1}z - (s')^{-1}z' = ((s')^{-1} - s^{-1}).r.x \bmod q.$$

Basing on this equation the secret key x will be calculated by adversaries as follow:

$$x = r^{-1}\left(s^{-1}.z - s'^{-1}z'\right)\left(s'^{-1} - s^{-1}\right)^{-1} \bmod q \quad (2)$$

Note: If the session key is coincided then the first component of the two signatures are coincided, but the inverse is not certain to be correctness.

The time complexity of the DSA signature scheme:

Let C_G is the time complexity of a signature generation algorithm, let C_V is the time complexity of a signature verification algorithm. we assume that the time complexity of the multiplication of two integer in $\mathbb{Z}_P$ is M_L and the time complexity of the multiplication of two integer in $\mathbb{Z}_q$ is M_N. Then analysis results is as follow:

The time complexity of the algorithm 1 focuses on formula: $g^k \bmod p$ and an inverse operation in Z_q, denoted $k^{-1} \bmod q$ and two multiplication operation in Z_q. According to [20, p176], if $len(p) = L$ and $len(q) = N$ then the time complexity of $g^k \bmod p \approx O(\log k.L^2) \approx N.M_L$ and the time complexity of $k^{-1} \bmod q \approx N.M_N$, thus the time complexity of the algorithm 1 is estimated as below:

$$C_G \approx N.M_L + (N+2)M_N \quad (3)$$

The time complexity of the algorithm 2 focus on the formula on step 5: $v \leftarrow (g^{u_1}.y^{u_2}) \bmod p) \bmod q$, it includes one multiplication operation and two exponentiations in Z_P. According to [20], the time complexity of the algorithm 2 is estimated as:

$$C_V \approx NM_L + (N+3)M_N \quad (4)$$

3. The proposed scheme

3.1. Generate Parameter

Let $n = p.q$, where p, q is the two distinct odd primes; $t = p_1.q_1$, where p_1, q_1 are two distinct odd primes, let p_1 is a divisor $p - 1$ that is denoted $p_1|(p-1)$, and $q_1 \mid (q-1)$.

Let p_1 isn't a divisor $q - 1$ that is denoted $p_1 \nmid (q-1)$, and $q_1 \nmid (p-1)$; an element g from Z_n^* and the order of g is t which is denoted by $ord_n(g) = t$; x is chosen randomly in $(1, t-1]$ and $\exists x^{-1} \bmod n$; $y = g^x \bmod n$; The signer's secret key is (n, g, x, t) and the public key is (n, g, y, N), where $N = len(t)$;

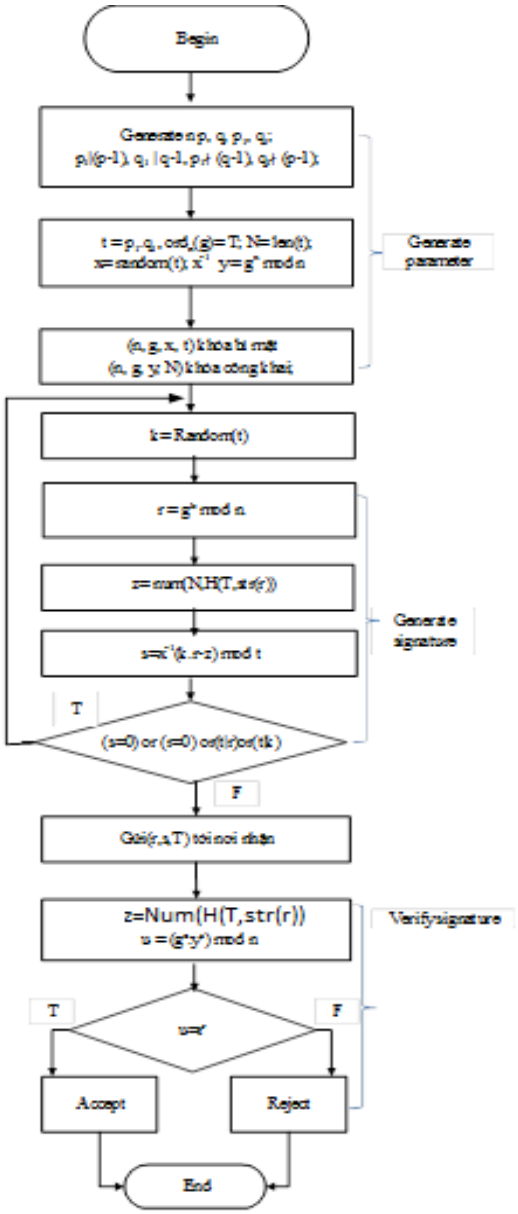

Fig. 1. Algorithm chart of proposed scheme

3.2. Generation signature and verification signature

Algorithm 3: Generate signature

Input: $(n, g, x^{-1}, t), T \in \{0,1\}^*$

Output: (r, s).

1. $k \in_R (1, t)$.

2. $r \leftarrow g^k \bmod n$.

2. $z \leftarrow Num(H(T||Str(r)))$

3. $s \leftarrow x^{-1}.(k.r - z) \bmod t$.

4. $if\ (s = 0) or\ (p_1|k) or\ (q_1|k) or\ (t|r)$ then goto 1.

5. return (r, s).

Algorithm 4: Signature verification

Input: T, (r, s), (n, g, y, N).

Output: "*accept*" hoặc "*reject*".

1. $z \leftarrow \boldsymbol{Num}(\boldsymbol{H}(T||str(r)))$.

2. $u \leftarrow g^z . y^s \ mod \ n$

6. *if* $(r^r = u)$ return "*accept*" else return "*reject*".

Proof of Correctness:

It's easy to see that:

$u = (g^z . y^s) mod \ n =$

$= g^z . g^{(x.x^{-1}rk - z.x.x^{-1}) mod \ t} mod \ n$

$= g^{k.r} mod \ n = r^r$

The proposed scheme is illustrated in Fig 1.

3.3. Analysis signature scheme

There are some differences between our scheme with the DSA scheme and the Elgamal scheme such as: In our signature scheme, the modular number n is composite that calculated by $n = p.q$, where p, q is the two distinct odd primes, in addition the order of the primitive element g (denoted by $ord_n(g) = t$), t is composite and be kept secret. Meanwhile the DSA scheme's the modular number p is prime and $ordp(g) = q$, (q is a primes) and be not kept secret.

Security analysis:

This section will show why using an DLP_n-based digital signature scheme is secure than using a digital signature scheme on DLP_p. Some situations are considered as follow:

The first situation:

The session key is revealed, the secret key x is calculated by the following formula:

$s \leftarrow x^{-1}(k.r - z) \bmod t \ \rightarrow \ x \leftarrow s^{-1}.(k.r - z) \bmod t$. Because the value of t is kept secret, in order to recover value t, adversaries have to face the difficulty of solving order problem (OP) in ring Z_n.

The second situation: Suppose that T and T' are two signed messages that are coincided the session key $k, k \in Z_t$, then the first component of two signatures is calculated by $r = g^k \bmod n$, r is same for both signatures, but the second component of two signatures are different and denoted by s, s'.

$z \leftarrow Num(H(T||str(r)))$

$z' \leftarrow Num\ (H(T'||str(r)))$

$s \leftarrow x^{-1}(k.r - z) \bmod t \ \leftrightarrow \ k = (s.x + z).r^{-1} \bmod t$

$s' \leftarrow x^{-1}.(k.r' - z') \bmod t$

$\leftrightarrow k = (s'.x + z').r^{-1} \bmod t$

$x = (z'r^{-1} - z.r^{-1})^{-1}(s.r^{-1} - s'.r^{-1}) \bmod t.$ (5)

Because the value of t is kept secret, and if someone want to recover value of t, he has to face the difficulty of solving order problem (OP) in ring Z_n.

The third situation: if an adversary gain the signer's secret key x then he can't forge our signature scheme's the signature. Suppose that a adversary recover the secret key x of a user, in order to forge our scheme's the signature, at first he has to calculate the first component of signature that is calculated by $r = g^k \bmod n$, after that he can calculate the second component of signature s is as follow:

$z \leftarrow Num(H(T||str(r)))$

$s \leftarrow x^{-1}(k.r - z) \bmod t$ (6)

But value of t is kept secret, thus the value s can't be calculated and the signature can't be forged by adversaries.

In short, our scheme's the first component of the signature is hashed together with the message, in addition the order of the primitive element g (denoted by $ord_n(g) = t$), t is kept secret, therefore it overcame the disadvantages of the DSA scheme and the Elgamal scheme. Furthermore, basing on Chik How Tan's proof results [15], our scheme is secure against existential forgery under adaptive chosen-message attack relative to the hardness of DL problem under the random oracle model.

The time complexity of the proposed scheme:

Suppose that p, q are the prime numbers that are used for the proposed scheme. Let $L = len(n)$ Let $N = len(t), t = p_1.q_1$. Let C_G is the time complexity of a signature generation algorithm, let C_V is the time complexity of a signature verification algorithm. Assuming that the time complexity of the multiplication in $\mathbb{Z}_n$ is M_L and the time complexity of the two integer multiplication in $\mathbb{Z}_t$ is M_N. The time complexity of the algorithm 3 focuses on $y = g^k \bmod n$ and x^{-1} has been calculated previously. According to [20, p 176], the time complexity of the algorithm 3 is estimated as follow:

$C_G \approx NM_L + 2M_N$ (7)

The time complexity of algorithm 4 is mainly focused on the time complexity of exponentiation operation $g^{zs}.y^s$ in Z_n. Where

$g^{s.z}.y^{s} \bmod n = g^{s.z}.g^{s.x} \bmod n$. Then the time complexity of algorithm 4 is estimated as follow

$$C_v \approx 2NM_L + (N+2)M_N \quad (8)$$

In term of memory space complexity:

Assume in signature scheme is used by K members, each member uses a separated modulo number (in order to prevent attacking from using the same modulo number). Each the signature of two scheme (DSA and the proposed scheme) require $2N$ bit, in which $N = len(q)$ with the DSA, $N = len(t)$ with the proposed scheme (in practice N is chosen as follow: 160, 224, 256). At this point, the space complexity of the proposed scheme is greater than K times the DSA scheme.

4. Testing simulation

In this testing, setting the length of modular are 1024, 1280, 1536, 1792, 2048 (bit). The prime numbers of simulation testing are generated by the algorithm [21] [22]. The message's size of the testing is 25.87 MB. The test PC hardware confguration is: CPU Intel(R) Core (TM)2/3.00 GHz, the physical memory 2G byte and the operating system is Windows XP Professional. The Hash function SHA 512 is used for testing. The results of testing are shown in Table 1.

Table 1. Results of testing

Key size(bit)	Generation time		Verification time	
	DSA	**New**	**DSA**	New
1024	1.416	1.539	6.836	9.527
1280	1.814	1.953	10.765	14.931
1536	2.89	3.182	14.813	20.03
1792	3.5	4.916	19.18	25.13
2048	5.138	5.929	22.59	32.398

In order to compare the signing speed between the DSA scheme with the proposed scheme, we simulated the experimental results of Table 1 by graph (Fig 1, Fig 2).

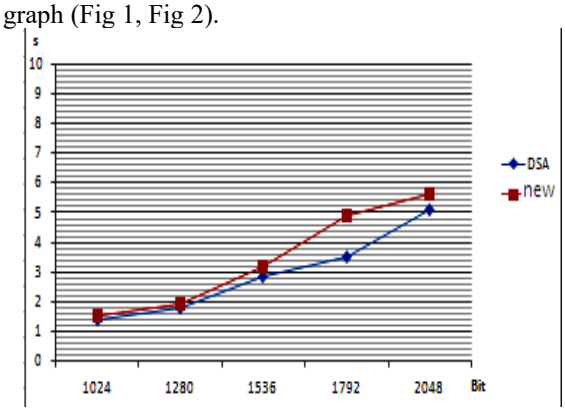

Fig. 2. The relationship between the size of the key and the signing time

Similarly, based on Table 1, the relationship between the wasting time and key size in order to verify the digital signature scheme DSA and the new digital signature scheme is depicted by graph as follow:

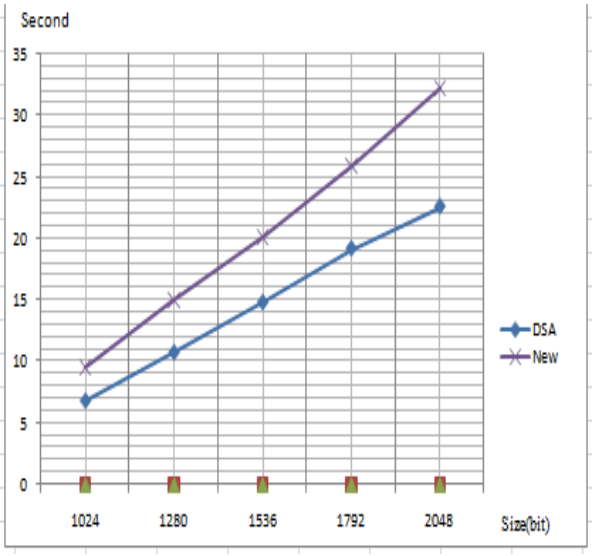

Fig. 3. The signature verification graph of the DSA scheme and the proposed scheme

5. Conclusion

The proposed scheme's security based on the discrete logarithm problem on ring Z_n in which the number of modulo n is a product of two primes. Because of this reason, in our scheme, the order of primitive element g that is denoted by $ord_n(g)$ can be kept secret; therefore our scheme is secure against situations of revealing or coinciding of session key. Furthermore, the security of proposed scheme is based on DLP_n that is considered to be more difficult than the DLP_p because someone want to solve it, he has to solve three problems, such as: FP, DLP_p and DLP_q. In addition, our scheme's the first componant of the signature is hashed together with the message. Thank to these differences, our proposed scheme is more secure in comparison with the best-known schemes such as: the DSA and Elgamal scheme, therefore it can be prevented attacks from revealing or coinciding session key. Furthermore, in our scheme, the inverse element of the secret key (denoted by $x^{-1} \bmod t$) is pre-calculated, so the complexity of signature generation is similar to the complexity of signature generation in the DSA scheme that is suitable for smart cards. However, there may be an attacking method applying for the proposed scheme that has never been known, this is also the need for further study.

Reference

[1]. T. ElGamal. A public key cryptosystem and signature scheme based on discrete logarithms, IEEE Transaction on Information Theory. 1985, IT-31(4): pp. 469 - 472.

[2]. W. C. Kuo, On ElGamal Signature Scheme, Future Generation Communication and Networking (FGCN 2007*)*, Jeju, 2007, pp. 151-153

[3]. C. P. Schnorr, Efficient signaturegeneration for smartcards, Journal of Cryptology Vol. 4, pp. 161-174, 1991.

[4]. T. S. Ng, S. Y. Tan and J. J. Chin, A variant of Schnorr signature scheme with tight security reduction, 2017 International Conference on Information and Communication Technology Convergence (ICTC)*,* Jeju Island, Korea (South), 2017, pp. 411-415.

[5]. H. Morita, J.C. Schuldt, T. Matsuda, G. Hanaoka, T. Iwata. On the security of the schnorr signature scheme and DSA against related key attacks. International Conference on Information Security and Cryptology - CRYPTOLOGY '15, pp. 20–35, Springer, 2015.

[6]. National Institute of Standards and Technology (NIST), FIPS Publication 186: Digital Signature Standards (DSS)(1994)

[7]. Sung-Ming Yen and Chi-Sung Laih, Improved digital signature algorithm, *in* IEEE Transactions on Computers, vol. 44, no. 5, pp. 729-730, May 1995.

[8]. Z. M. Chen. An inproved encryption algorithm on ELGamal algorithm, Computer Applications and Sostware, vol. 22. 2005, pp.82- 85.

[9]. J.-m.Liu,X.-g.Cheng,andX.-m.Wang, Methods to forge elgamal signatures and determine secret key, in Advanced Information Networking and Applications, 2006. AINA 2006.20th International Conferenceon, vol.1.IEEE, 2006, pp. 859–862.

[10]. L. Xiao-fei, S. Xuan-jing and C. Hai-peng, An Improved ElGamal Digital Signature Algorithm Based on Adding a Random Number 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing*,* Wuhan, Hubei, 2010, pp. 236-240.

[11]. Z. Ping, K. Yingzhan and J. Keke, Instruction-Cache Attack on DSA Adopting Square-Multiply Method, 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control, Harbin, 2012, pp. 905-908 6-11.

[12]. B. Yang, A DSA-Based and Efficient Scheme for Preventing IP Prefix Hijacking, 2014 International Conference on Management of e-Commerce and e-Government, Shanghai, 2014, pp. 87-92.

[13]. Z. Ping, W. Tao and C. Hao, Research on L3 Cache Timing Attack against DSA Adopting Square-and-Multiply Algorithm, 2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC), Qinhuangdao, 2015, pp. 1390-1393.

[14]. M. Girault, An identity-based identification scheme based on discrete logarithms modulo a composite number in Advances in Cryptology - Eumcrypt'SO, Lecture Notes in Computer Science 473, Springer-Verlag, pp.481-486, 1991.

[15]. Chik How Tan, Xun Yi and Chee Kheong Siew, Signature scheme based on composite discrete logarithm, Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint, 2003, pp. 1702-1706

[16]. S. K. Tripathi and B. Gupta, An efficient digital signature scheme by using integer factorization and discrete logarithm problem, 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*,* Udupi, 2017, pp. 1261-1266.

[17]. E. Okamoto and K. Tanaka, Key distribution system based on identification information, in *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 4, pp. 481-485, May 1989.

[18]. Boyd, C. Digital signature and public key cryptosystem in a prime order subgroup of Z_n^*. First International Conference on Information and Communications Security, ICICS' 97 (LNCS1334), pages 346-355.Springer,1997.

[19]. E. Okamoto and K. Tanaka, Key distribution system based on identification information, in *IEEE Journal on* Selected *Areas in Communications*, vol. 7, no. 4, pp. 481-485, May 1989.

[20]. D.R Stinson, Cryptography Theory and Practice", CRC Press, pp 176, 2003

[21]. Tuan Le Van, Truyen Bui The Building a method for deterministic prime generation, The research journal of military science and technology, No.42, 04- 2016, ISSN 1859 – 1043.

[22]. Richard Crandall, Carl Pomerance. Prime Numbers, A Computational Perspective, Second Edition, Springer Science + Business Media, Inc, 2005