DOI: 10.13140/RG.2.2.12917.99044

Study of Presentation Confidence Improvements in ECC Algorithm with Exposure and Preclusion of Framing and Forgery Attacks

Saumya Rajvanshi[1], Gurjot Singh Sodhi[2]
[1]M.Tech (Student), [2]Assistant Professor
Shaheed Udham Singh College of Engineering and Technology, Tangori, Mohali

Abstract **-** Elliptical curve cryptography is an unrestricted key encryption approach based in elliptic curve concept that can be used to generate speedily, smaller and more efficient cryptographically keys. ECC creates keys through the properties as the product if the elliptic curve equation instead of the new method of creates as the product of very large prime numbers. Today we can find elliptic curves cryptosystems in Models which are just three in major technologies on which the newest web and information world are based. Before ECC become popular, almost all public key algorithms were based on RSA, DSA and DH alternative cryptosystems based in arithmetic. The problem or research work in which we are going to continue our work of Elliptic Curve Cryptography based Digital Signature Schema. The Protected transmission using elliptic curve cryptography is established on the encryption and decryption, they are most normally used in video conferencing, privacy of social media other data and they are the well-organized one that deal with the covertly of information and are most usually used to analyse, find the approaches for security of data. The Aim of my research work is to implement and compare the behaviour of improved CCH1 and CCH2 proxy multi-signature schemes based on ECC on the basis of time complexity, computational overhead and space complexity.

Keywords **-** Elliptical curve cryptography, RSA, DSA and DH, improved CCH1 and CCH2 proxy multi-signature schemes.

I. INTRODUCTION

In ECC a 160-bit key offers the similar security as related to the out dated crypto system RSA with a 1024-bit key, thus lesser the computer power. Consequently, [1] ECC gives significantly greater security for a given key size. Therefore, a key with small size makes it possible much more condensed executions for a given level of security which means faster cryptographic actions, running on smaller chips or more compact software. Additional, there are particularly efficient, compact hardware executions are available for ECC exponentiation operations, subscription potential reductions in application footprint even beyond those due to the shortest key extent alone. Elliptic curve cryptography is not just appeared as an good-looking open key crypto-system for mobile or wireless atmosphere but also gives bandwidth savings. Elliptic curve cryptography is not easy to apprehend by attacker. The Security Study i.e [2];

1. Key space

Security of a cryptographic approach depends a set on the dimension of the key used. The method will be recognized to all. It is always a good special to have a big key size but we should also keep in attention the computational load when we increase the key size. ECC delivers a computationally hard problematic called Elliptic Curve Discrete Logarithmic. Problematic which helps in using a smaller key size associated to other cryptographic procedure and still holds the safety level high due to ECDLP. In our operation we have used a 192 bit key length, which is quite decent to keep against naive attack. For improved security we could raise the key distance used for encryption & decryption [3].

1. Key Sensitivity

A small change in the unique key should produce a totally dissimilar improved message.

2. Cipher text only attack

Given that the cryptanalyst knows the encryption procedure and the cipher text. Pending and unless, the cryptanalyst does not have the remote key of the receiver the attacker can't expose the plain text. Applying Brute Force attack would not be of considerable help while that key size is very big as it will take a lot of time in term of years. So, even if the expert is able to decrypt it, the value of the information will be no more at that time.

3. Known plaintext attack

Assumed that the cryptanalyst identify the encryption method, cipher text and one or more plaintext-cipher text pairs formed with the secret key. Since, the implementation generates a dissimilar cipher text for the same message due to the random *k* used in the procedure.

II. CRYPTO-SYSTEMS AND PUBLIC KEY CRYPTOGRAPHY

The expression "Cryptography" is derived from the Greek & it literally way "secret script". Cryptography has been about for more than a thousand years and the Roman territory was thought to[4] be the main of cryptography as they used simple cipher methods to hide the meaning of messages. Some of the earlier and popular cryptographic

DOI: 10.13140/RG.2.2.12917.99044

methods were Caesar cipher, Replacement cipher and Transposition ciphers. Cryptography is the process of encoding the plain text into an unintelligible cipher text by the process of Encryption and the conversion back to plain text by process of Decryption[5].

Cryptographic systems are usually classified on the following origin:

1. **Type of procedures used to for converting plaintext to cipher text**: mainly encryption algorithms are based on 2 common main beliefs,

a. ***Replacement***, in which each component in simple text is mapped to some further part to form the cipher text

b. ***Transposition,*** in which components in plaintext are rearranged to form cipher text.[6]

2. **Number of keys used**: If together the sender and the receiver use a same key then such a method is referred to as Symmetric, single-key, secret-key or conservative encryption. If the sender and receiver use dissimilar keys, then such a scheme is called Asymmetric, Two-key, or public-key encryption.
3. **Processing of Plain text**: A Block cipher procedure the input one block at a moment, producing an productivity block for each input block. A flow cipher processes the input elements incessantly constructing output elements on the fly.

Most of the cryptographic procedures are either symmetric or asymmetric key algorithms.

1. **Top-secret Key Cryptography:** This nature of cryptosystem uses the similar key for both encryption & decryption. Some of the compensations of such a system are
   - Very fast comparative to communal key cryptography
   - Considered secure, as lengthy as the key is strong

   Symmetric key crypto-systems have some difficulties too[8]. Exchange and administration of the key becomes complex. Non-repudiation is not probable. Some of the examples of Symmetric key cryptosystems contain DES, 3-DES, RC4, RC5 etc [7].

2. **Public Key Cryptography:** This type of cryptosystems uses dissimilar keys for encryption & decryption. Each user has a public key, which is branded to every others, & a private key, which leftovers a secret. The private key and public key are statistically linked. Encryption is perform with the public key & the decryption is performed with the private key. Public key cryptosystems are measured to be very secure & supports Non-repudiation. No switch of keys is compulsory thus reducing key organization to a smallest. But it is much slower than Symmetric key method and the encryption text tend to be much larger than plaintext. Some of the exemplar of public key cryptosystems comprise Diffie-Hellman, RSA and Elliptic Curve Cryptography [8].

III. CRYPTANALYTIC ATTACK

ECC one time we have a large digit of keys to frustrate a brute-force attacker, we need to create sure no shortcuts are accessible to an analytical attacker. Currently, no sub exponential algorithms are known to solve the general elliptic curve discrete logarithm predicament (ECDLP). According to , the best known attack to date on the universal ECDLP is the Pollard ρ–way. Let n be the arrange of the given point (the smallest number of times for which the peak must be added to itself to get 0) on the curve. The Pollard ρ–method, based on the principle of the so-describe "birthday problem," takes about (πn/4)½ elliptic curve additions. This is on the order of the amount of work desirable to perform an exhaustive key explore. However, certain types of elliptic curves have structures that make the ECDLP solvable in sub exponential moment. Apparently, the goal is to avoid those particular types of curves when selecting an suitable curve. Below are the well-known attacks alongside ECDLP on some specific curves according to[9]:

a) Forgery attack

1. Forge the substitute signer's signature :

After getting signature (m, m_w, R, e, y) the original signer $A_1 \ldots A_n$ can forge proxy signer P's signature on message m as follows:

Where $e = h(m, j_x)$ and $y = j - de \pmod t$

- Each A_i compute $s_i * e$ where i=1,..n;
- Compute $y' = \sum_{i=1}^{n} s_i * e \bmod t$;
- Compute $y'' = y + y'$;
- (m, m_w, R, e, y'') is valid signature on message m .

Then the malicious original signers can forge a valid signature (m, m_w, R, e, y'') on message m respect to proxy signer P's public key d_p.[10]

The following shows why the signature (m, m_w, R, e, y'') is valid

PROFF:

$y'' = y + y'$

$= j - de + \sum_{i=1}^{n} s_i * e \bmod t$

$= j - (d_p + \sum_{i=1}^{n} s_i)\, e + \sum_{i=1}^{n} s_i * e \bmod t$

$y'' = j - d_p\, e \bmod t$

2. Forge a proxy multi-signature

Suppose proxy signer P signed a message with his private key d_p , the signature is (m, m_w, R, e, y'') where $e = h(m, j_x)$ and $y = j - d_p e \pmod t$.

DOI: 10.13140/RG.2.2.12917.99044

Upon received the signature (m, m_w, R, e, y´´), then the malicious original signers $A_1 \ldots A_n$ can forge a valid proxy signature as follows:

- Compute y´=$\sum_{i=1}^{n} s_i$ * e mod t.
- Compute y= y´´ - y´

Finally , the malicious original signers $A_1 \ldots A_n$ can forge a valid proxy signature (m, m_w, R, e, y).[10]

The following shows why the proxy signature (m, m_w, R, e, y) is valid.

PROFF:

y= y´´ - y´

= j - d_p e mod t -$\sum_{i=1}^{n} s_i$ * e mod t

= j- (d_p+$\sum_{i=1}^{n} s_i$) e mod t

y= j – de mod t

b) Framing Attack

In this attack, malicious users A_1, $A_2 \ldots$ A_n also can forge a proxy multi-signature for message m by some user P on behalf of users A_1, $A_2 \ldots$ A_n, such that user P was never designated by users $A_1, A_2 \ldots A_n$.

Suppose proxy signer P signed a message with his private key d_p, the signature is (m, m_w, R, e, y´´) where e= h(m, j_x) and y = j-d_pe (mod t).

Upon received the signature (m, m_w, R, e, y´´), then the malicious original signers $A_1 \ldots A_n$ can forge a valid proxy signature as follows[11]:

- The malicious users $A_1, A_2 \ldots A_n$ pretend to produce a forge warrant m_w,which recording the delegation information such as identities of the malicious users A_1-,$A_2 \ldots A_n$ and user P .
- For each 1≤ i ≤n, the malicious user $A_i$selects a random number 1≤ k_i≤ t– 1, and then computes R_i= k_iX B = (x_{Ri},y_{Ri}) and broadcast R_i to other users
- On receiving R_j(1 ≤ j ≤ n, j ≠ i) , $A_i$calculates

$$R= \sum_{i=1}^{n} R_i=(x_R,y_R)$$

$$s_i= d_i * h(M_w,x_{Q_p,}x_{Q_i}x_R) – k_i x_R(\text{mod } t),$$

Note that user P doesn't receive any information from the malicious users $A_1, A_2 \ldots A_n$.[12]

- Compute y´=$\sum_{i=1}^{n} s_i$ *e mod t;
- Compute y= y´´- y´

Finally the malicious users can forge a valid signature (m, m_w, R, e, y) on message m by some user P on behalf of users $A_1 \ldots A_n$, such that user P was never designated by users $A_1 \ldots A_n$. The following shows why the signature (m, m_w, R, e, y) is valid.

PROFF:

y= y´´- y´
= j-d_pe mod t -$\sum_{i=1}^{n} s_i$ *e mod t
= j-(d_p +$\sum_{i=1}^{n} s_i$) e mod t
= j- de mod t

From above we can see that an innocent user P be framed by the malicious users $A_1 \ldots A_n$.

IV. RELATED WORK

Aditya Babel et al. in 2010[13] explored the basic structures of elliptic curve cryptography without going into the complicated mathematical details. They advance some mathematical theory in describing elliptic curve groups and their internal operations. Throughout this paper, they compare ECC to other asymmetric encryption structures such as RSA and ELgamal and, in doing so, hope to influence the reader that, despite its somewhat disgusting and complicated look, ECC is indeed a consistent cryptographic scheme that will be significant in the near future.

Haodong Wang et al. in 2006 [14] described a public key implementation of access control in a sensor network. They detail the implementation of Elliptic Curve Cryptography over primary field, a public-key cryptography scheme, on TelosB, which is the modern sensor network platform. They appraise the performance of implementation and compare with other operations we have ported to TelosB.

Ikshwansu Nautiyal et al. in 2012 [15] described as Cryptography is the method of hiding a message in some indecipherable format so that the message lies hidden in plain sight of an accidental person. The methods of cryptography are centuries old. With technical development, techniques have evolved knowingly. Public key cryptography offers a wide variety of security over the various modes of transporting data, especially over Internet. The security of a public key encryption is stronger only if the validity of the public key is ensured. Data encryption values like RSA and Diffie- Hellman are becoming incompetent due to requirement of large quantity of bits for cryptographic process. As of newest, ECC has become the latest trend in the cryptographic situation. This paper presents the operation of ECC for encryption or decryption and confirmation process, using JAVA as the implementation tool.

K.S. Abitha et al. in 2015[16] described as protected data transmission using elliptic curve cryptography can be well-defined as transmission of data. This paper suggested a review about Secured data transmission using elliptic curve cryptography. The main problematic in present system is safety issues in transmitting data between foundation and

DOI: 10.13140/RG.2.2.12917.99044

the destination. After the review on various literature papers, they are concluding a new way that increases security deliberations of the network using AODV algorithm for transmission of data and to increment the efficiency of AODV algorithm using Elliptic Curve Cryptography.

Laiphrakpam Dolendro Singh et al. in 2015 [17] described as elliptic curve cryptography has been a latest research area in the field of Cryptography. It offers higher level of security with smaller key size associated to other Cryptographic methods. A new method has been suggested in this paper where the standard procedure of planning the characters to affine points in the elliptic curve has been detached. The analogous ASCII values of the plain text are matching up. The opposite values serve as contribution for the Elliptic curve cryptography. This new method avoids the expensive operation of planning and the need to share the shared lookup table between the sender and the receiver.

V. SIMULATION MODEL

To implement and compare the behaviour of improved CCH1 and CCH2 proxy multi-signature schemes based on ECC on the basis of time complexity, computational overhead and space complexity. To cryptanalyze improved CCH2 proxy multi-signature scheme by applying various attacks like Original signer forgery attack, Transferring attack, framing attack. To propose and implement improved CCH2 proxy multi-signature scheme based on ECC for handling above attacks.

- System initialization phase:

1: **A** field size q, which is a large odd prime,$q \approx 2^{160}$ bit integer.
2: Two parameters a,b ε $F_q$to define the equation of elliptic curve E over F_q, (i.e., $y^2 = x^3 + ax + b \pmod p$ in the case p > 3), where $\mathbf{4a^3} + 27\ b^2 \neq \mathbf{0} \pmod p$
3: A finite point B = **(x_B, y_B)** whose order is a large (160-bit) prime number in $E(F_q)$, where B is a point in $E(F_q)$ .where $B \neq O$, because O denotes an infinity point.
4: The order of B = t.

Declare (q, a, b, B, t) publicly so that a verifier can refer these parameters to verify.

5. For each $1 \leq i \leq n$, the original signer A_i secretly selects a random number $1 \leq d_i \leq t - 1$ as his private key, and computes the corresponding public key $Q_i = d_i X\ B = (x_{Q_i}, y_{Q_i})$, where "X" indicates the multiplication of a number by an elliptic curve point[13].
6. Let h( ) be a public collision-resistant hash function that must be secure.
7. Then, the proxy signer is provided with a private key $1 \leq d_p \leq t - 1$ and a corresponding public key $Q_p = d_p X\ B = (x_{Q_p}, y_{Q_p})$.
Where n is no. of original signer.

Sub proxy key generation:

1. For each $1 \leq i \leq n$, the original signer Ai selects a random number $1 \leq k_i \leq t - 1$, and then computes $R_i = k_i X\ B = (x_{Ri}, y_{Ri})$ and
2. If $x_{Ri} = 0$ then return to step 1; otherwise A_i broadcasts $R_i$to other original signers.
3. On receiving $R_j (1 \leq j \leq n, j \neq i)$, $A_i$calculates

$$R = \sum R_i = (x_R, y_R), \text{ where } i = 1,2 \ldots n.$$

$$s_i = d_i * h(M_w, x_{Q_p}, x_{Q_i} x_R) - k_i x_R \pmod t,$$

where $M_w$is a warrant that includes the originalsigners' ID, the proxy signer's ID, then delegation period[14].
and sends $\sigma_i = (m_w, R, s_i)$ to the proxy signer via a public channel.
4. On receiving σ_i from $A_i$for $1 \leq i \leq n$, the proxy signer P checks whether

$$S_i\ B = h(M_w, x_{Q_p}, x_{Q_i} x_R) * Q_i - x_R R_i$$

holds. If it holds, σ_i is valid; otherwise, the scheme fails.
5. If all σ_i 's are valid, then B calculates

$$d = d_p + \sum s_i \bmod t$$

as the proxy signing key[13].

Proxy signature generation

The proxy multi-signature affixed to the m is in the form of $(m, m_w, R, Sig_d(m))$, where $Sig_d(m)$ is the signature generated by a EC-schnorr signature scheme using the proxy signing key d. where m is message say HELLO.

1. Proxy signer P choose random number m where $1 \leq j \leq t - 1$ and calculate $j_o = j * B = (j_x, j_y)$
2. Compute $e = h(m, r_x)$ where $h(r_x, m)$ is hash function: $F_q \times \{1,0\}^{|m|} \rightarrow f_n$

If e = 0 then goto step 1;
3. Compute y = j-de (mod t)
And the output $sign_d(m) = (e, y)$.

Proxy signature verification

When the verifier verifies the signature, he or she calculates the proxy public value Q corresponding to the proxy signature key d as

$$Q = Q_P + \sum h(M_w, x_{Q_p}, x_{Q_i} x_R) * Q_i - x_R\ R$$

With the value, the verifier can confirm the validity of $Sig_d(m)$ by validating the verification equality of the designated signature scheme.
Compute $\bar{J}_o = yB + eQ = (\bar{J}_x, \bar{J}_y)$
And compute $e' = h(\bar{J}_x, m)$
Check that $e' = e$ and if these are equal then valid signature otherwise not.

Process 1: System initialization phase: Before the whole scheme can be initialized, the following parameters over the elliptic curve domain must be known.

Process 2: Key generation phase: This phase can be further divided into two parts.

DOI: 10.13140/RG.2.2.12917.99044

Part 1: Personal public key generation phase: All original signers and the designated proxy signer are authorized to select their own individual secret keys.

Part 2: Proxy-signature secret key generation phase.

Process 3: Proxy multi-signature generation phase: The proxy multi-signature affixed to the m is in the form of $(m,m_w,R,Sig_d(m))$, where $Sig_d(m)$ is the signature generated by a designated signature scheme (ECDSA) using the proxy signing key d and m is message.

Process 4: Proxy multi-signature verification phase: When the verifier verifies the signature, he or she calculates the proxy public value Q corresponding to the proxy signature key d.

VI. SIMULATION RESULTS

We described the result and comparison of the performance parameters between previous work and proposed work.

Inputs	CCH	Proposed
3,7	204	161
7,11	210	177
11,13	202	163
13,17	206	174

Here the results for number of operations for generation keys and their verifications in the ECC environment. This process shows that high number of operations cause high memory consumption and processing time as well. In current scenario the proposed method optimize the key generation and verification of ECC and gives the better results as shown in the table and graph as well below:

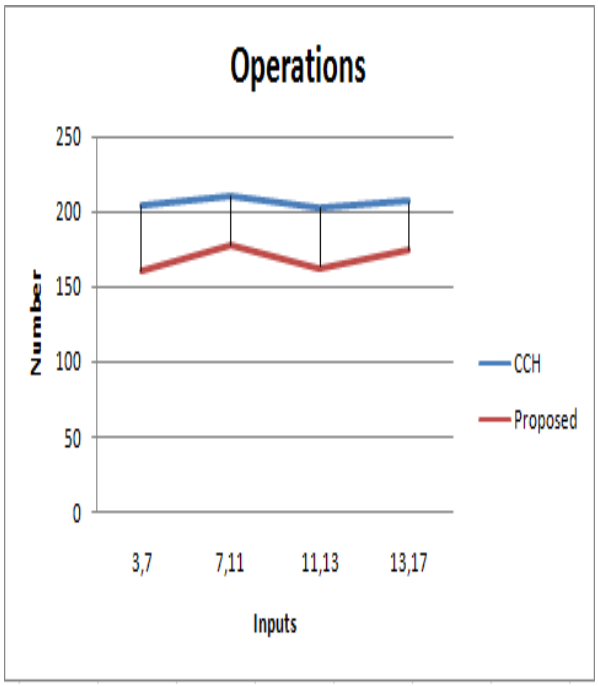

Fig.1: Number of Operations

Input	CCH	Proposed
3,7	1986	1349
7,11	1513	1404
11,13	1553	1488
13,17	1793	1421

This table shows that the total number of bytes consumption by the algorithms and their comparison with the previous one. The more usage of memory cause low speed and high chances of system crashes. So here the performance of proposed technique is better than the previous one. The comparison between them is as:

DOI: 10.13140/RG.2.2.12917.99044

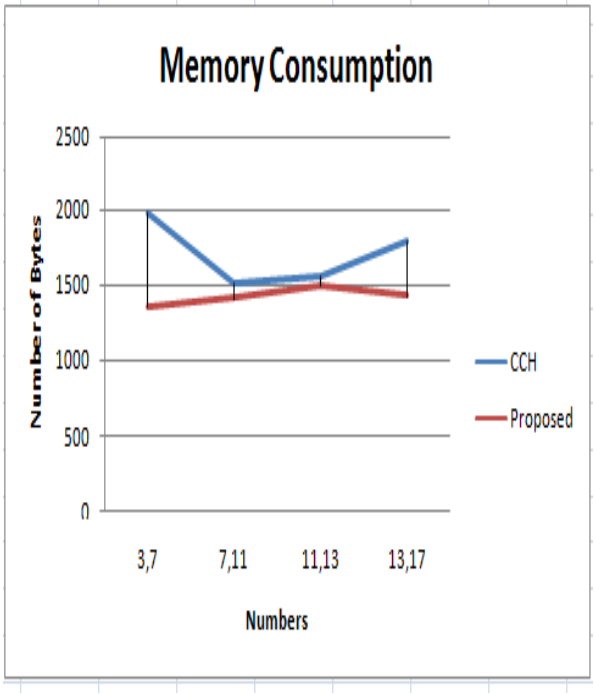

Fig.2: Number of Bytes (Space Consumption)

Inputs	CCH	Proposed
3,7	1723	1373
7,11	1724	1402
11,13	1788	1378
13,17	1718	1418

This process shows the total time taken via algorithm to complete the process of key generation and verification process. The time consumption shows the efficiency of algorithms in real time environment. Here the number of inputs varies but the results are still improved with the use of proposed technique. The comparison in graph is as:

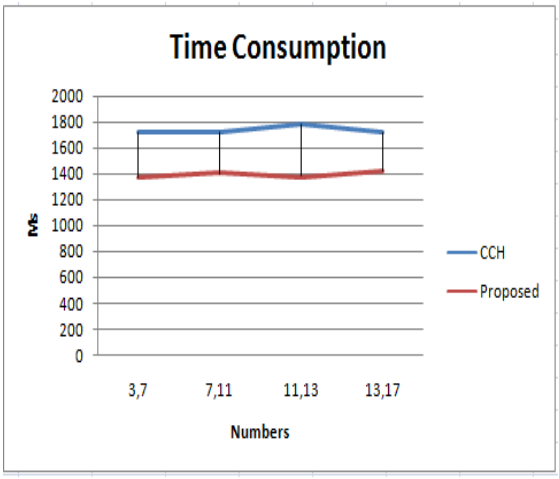

Fig.3: Time Consumption (Ms)

All the processes show that the results of the proposed technique are better than the CCH ECC key generation technique. Also the proposed technique detects and prevents various attacks in ECC Key generation environment. It maximizes the security of transmission and storage of data. The detection and prevention structure works with framing and forgery attacks in this process. Here verification snapshots for both process to show the efficiency of algorithms to working in critical environment.

VII. CONCLUSION ANF FUTURE SCOPE

We improved CCH2 proxy multi-signature scheme procedure for transfer of statistics and to increment the efficiency of algorithm using ECC (Elliptic Bend Cryptography). Efficiency, and dependability will be augmented for each transmission of data, While encircling the future method by using the ECC procedure which allow itself to encode and decrypt the data that is to be transported and performs the active organization, we are final that the Tenable data broadcast using elliptic bend cryptography provide a efficiency higher than vector when compared with previous work. To cryptanalyze improved CCH2 proxy multi-signature scheme by applying various attacks like Original signer forgery attack, Transferring attack, framing attack. So the data which is conveyed has to be encrypted and decrypted so that the security matters will be abolished and with the usage of the resources and effective distribution to the user. The future method will provide an effective solution to secure the information in elliptic curve cryptography that may help the source and terminus to transfer data in a secured way using encryption and decryption.

VIII. REFERENCES

[1]. V. Miller, “Uses of elliptic curves in cryptography”, Advances in Cryptology - CRYPTO'85, LNCS 218, pp.417-426, 1986.

DOI: 10.13140/RG.2.2.12917.99044

[2]. Kani, Ernst. "The State Of The Art Of Elliptic Curve Cryptography." Queen's University.
[3]. Kapoor, Vivek, Vivek Sonny Abraham, and Ramesh Singh. "Elliptic curve cryptography." Ubiquity 2008.May (2008): 7.
[4]. Afreen, Rahat, and S. C. Mehrotra. "A review on elliptic curve cryptography for embedded systems." arXiv preprint arXiv:1107.3631 (2011).
[5]. Gupta, Vipul, et al. "Performance analysis of elliptic curve cryptography for SSL." Proceedings of the 1st ACM workshop on Wireless security.ACM, 2002.
[6]. Allen, Jonathon Brandon. "Method and system for ensuring royalty payments for data delivered over a network." U.S. Patent No. 6,041,316. 21 Mar. 2000.
[7]. Tzeng, Wen-Guey. "A time-bound cryptographic key assignment scheme for access control in a hierarchy." IEEE Transactions on Knowledge and Data Engineering 14.1 (2002): 182-188.
[8]. ShanmugalakshmiDr.R., Prabu M., "Research Issues on Elliptic Curve Cryptography and Its applications", International Journal of Computer Science and Network Security, Volume 9, No.6, June 2009.
[9]. Yin, Edward. "Curve Selection in Elliptic Curve Cryptography." (2005).
[10]. Zhang, Ke-Jia, Wei-Wei Zhang, and Dan Li. "Improving the security of arbitrated quantum signature against the forgery attack." Quantum information processing 12.8 (2013): 2655-2669.
[11]. Li, Xiangxue, and Kefei Chen. "ID-based multi-proxy signature, proxy multi-signature and multi-proxy multi-signature schemes from bilinear pairings."Applied Mathematics and Computation 169.1 (2005): 437-450.
[12]. Kim, Jinsub, Lang Tong, and Robert J. Thomas. "Data framing attack on state estimation with unknown network parameters." 2013 Asilomar Conference on Signals, Systems and Computers.IEEE, 2013.
[13]. Aditya Birla, Alfred Menezes, and Scott Vanstone."The state of elliptic curve cryptography." Towards a quarter-century of public key cryptography.Springer US, 2010.103-123.
[14]. Wang, Haodong, Bo Sheng, and Qun Li. "Elliptic curve cryptography-based access control in sensor networks." International Journal of Security and Networks 1.3-4 (2006): 127-137.
[15]. IkshwansuNautiyalet.al,"Encryption using Elliptic Curve Cryptography using Java as Implementation tool", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 1, January 2014.
[16]. Karthikeyan, B., N. Kanimozhi, and S. Hari Ganesh. "ENCRYPT-SECURITY IMPROVED AD HOC ON DEMAND DISTANCE VECTOR ROUTING PROTOCOL (En-SIm AODV)." (2006).
[17]. Singh, LaiphrakpamDolendro, and KhumanthemManglem Singh. "Image Encryption using Elliptic Curve Cryptography." *Procedia Computer Science*54 (2015): 472-481.