ISSN 2250-3153

An Efficient Privacy Preserving and Authentication based Data Transmission Using Group Key Authentication Based on Secret Sharing

V. Naga Malleswara rao [*], D.Sreelakshmi[**],

PG Scholar, Dept. of C.S.E., Prasad V. Potluri Siddhartha Institute of Technology, Vijayawada, India
Asst. Professor, Dept. of C.S.E., Prasad V. Potluri Siddhartha Institute of Technology, Vijayawada, India

Abstract- In this paper we are proposing an improved privacy preserving and authentication based data transmission over network with ECDSA and QR vector model. QR Vector model generate quotient and reminder vectors for the sender data and digital signatures applies over vectors and signature verified at the receiver end and vectors can be converted to plain text with delta key value. Our proposed approach is more secure than the traditional approaches

Index Terms- Quotients and Reminder Model, Digital Signature, ECDSA, Bit coin.

I. Introduction

Elliptic curve cryptography needs the bit size and public key in twice the size of the level of security. Assume the level of security is 80 bits and the size of a digital signature algorithm public key and it has at least 1024 bits and the elliptic public key would be 160 bits.

Private key: A confidential number, known just to the individual that created it. A private key is basically an arbitrarily produced number. In Bit Coin, somebody with the private key that relates to supports on the public record can spend the trusts. In Bit Coin, a private key is a solitary unsigned 256 piece whole number (32 bytes).

Public key: A number that relates to a private key, however does not should be kept confidential. A public key can be computed from a private key, however not the other way around. A public key can be utilized to figure out whether a signature is authentic (as such, delivered with the correct key) without requiring the private key to be unveiled. In Bit Coin, public key are either compacted or uncompressed. Packed public keys are 33 bytes, comprising of a prefix either 0x02 or 0x03, and a 256-piece whole number called x. The more seasoned uncompressed keys are 65 bytes, comprising of consistent prefix (0x04), trailed by two 256-piece numbers called x and y (2 * 32 bytes). The prefix of a compacted key takes into consideration the y quality to be gotten from the x esteem.

Signature: A number that demonstrates that a marking operation occurred. A signature is numerically created from a hash of something to be marked, in addition to a private key. The signature itself is two numbers known as r and s. With the public key, a scientific calculation can be utilized on the signature to confirm that it was initially created from the hash and the private key, without expecting to know the private key. Signatures are either 73, 72, or 71 bytes in length, with probabilities roughly 25%, half and 25% separately, in spite of the fact that sizes considerably littler than that are conceivable with exponentially diminishing like hood.[1,2].

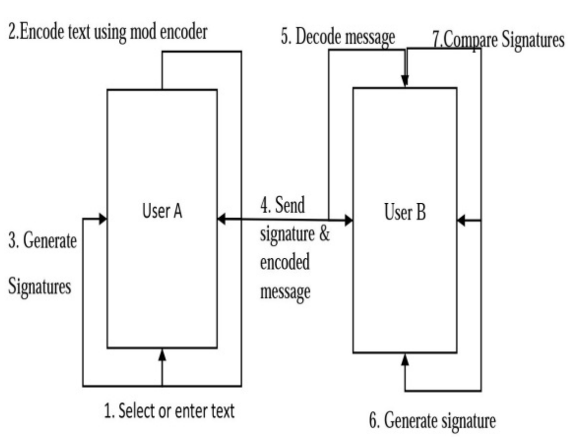
The quick development of Internet in the late days and the far reaching accessibility of systems have lead to the advancement of capable and innovative applications are getting to be on the web, also the Google Docs and Microsoft Office Live. Along these lines, the systems have turned out to be more transparent. The volume of information transmitted over the Internet is additionally expanding. In the blink of an eye, we have eBooks, sight and sound, e-business, open source applications, and so forth on the web. In this manner, the data on the Internet is turning out to be more touchy and powerless. Numerous applications request secret information correspondence between the sender and the beneficiary. In addition, such overpowering web activity requests the proficient utilization of data transmission accessible. Along these lines, we require secure correspondence with low data transfer capacity utilization. In such manner, the part played by information pressure gets to be critical as it offers an appealing methodology for decreasing the correspondence costs by utilizing the accessible transmission capacity adequately. Moreover, printed information, where each and every character matters, can't be stood to be compacted with Loss Compression methods.[3]

Symmetric Encryption also called as single-key encryption, one-key encryption and private key encryption. And is a type of encryption where the same secret key is used to encrypt and decrypt information or there is a simple transform between the two keys. A secret key can be a number, a word , or just a string of random letters. Secret key is applied to the information to change the content in a particular way. This might be a simple as shifting each letter by a number of places in the alphabet. Symmetric algorithms require that both the sender and the receiver know the secret key, so they can encrypt and decrypt all information.

III PROPOSED ALGORITHM

In our work the system we proposed a strong curve theory for generating strong signature for the message. We implemented encoding algorithm which is based on the secret value using mathematical operations. The procedure is explained below.The generation of the public key in ECDSA involves computing the point, Q, where $Q = dP$. In order to crack the elliptic curve key, adversary Eve would have to discover the secret key d. Given that the order of the curve E is a prime number n, then computing d given dP and P would take roughly $2^{n/2}$ operations [1]. For example, if the key length n is 192 bits (the smallest key size that NIST recommends for curves defined over GF(p)), then Eve will be required to compute about 2^{96} operations. If Eve had a super computer and could perform one billion operations per second, it would take her around two and a half trillion years to find the secret key. This is the elliptic curve discrete logarithm problem behind ECDSA.

Initially the elliptic curve theory algorithm is used to generate the signature for the encrypted text. The sequence as follows; The Lossless Mod-Encoder provides both encryption as well as compression on the data to be transmitted over the internet. It has the advantages over the compression mechanisms and symmetric key algorithms that is compression mechanisms does not provide the security to data and the symmetric key algorithms doesn't provide compression mechanisms. This algorithm uses a finite alphabet set, constant value Δ for encryption and a decryption of the message and is used as a secret key. This Δ is generated using Elliptic curve key generation algorithm to provide more security to algorithm. The sender generates Remainders and Quotients using Δ value and the compression performs only on the Quotient vector further these two values forwarded to the receiver to ensure the confidentiality of the message. The receiver decompresses and decodes the message using compressed quotient and remainder vector.



1. System Architecture

IV. PSEUDO CODE

L be a language, ∑ be an alphabet set, Data String M be $\{m_1, m_2, m_3 \ldots m_n\} \in \sum$. Δ is a constant value and is used for calculates quotients and remainders and this Δ is generated using any key generation algorithm. The quotient vector is also represented by using $B = \lceil |\sum|/\Delta \rceil + 1$.

Entity A performs the following steps to generate a public and private key:

1. Select an elliptic curve E defined over a finite field F_p such that the number of points in $E(F_p)$ is divisible by a large prime n.
2. Select a base point, P, of order n such that $P \in E(F_p)$
3. Select a unique and unpredictable integer, d, in the interval [1, n-1]
4. Compute $Q = d_p$
5. Sender A's private key is d
6. Sender A's public key is the combination (E, P, n, Q)

Generation of Delta Value:
If the key value < 29
Add 29 to the key
If the key value >=29 and <256
Assume the key value as Delta
If the key value >256
Calculate key mod 256

Encoding Algorithm:

1. Input : M ∈ ∑, Δ value
2. N=|M|, i.e length of M
3. Z=n * bit size, i.e bit size is the number of bits require to represent each

4. For i=1 to n
Read mi the i_{th} character from M
Find R
Find Q
Representation of R

5. For I=1 to n
Represent $R_{[i]}$ in base Δ
Representation of Q

After processing of the encoding algorithm we got two vectors that is reminder vector and Quotient vector.

Then generate the signature using elliptic curve signature algorithm using the generated key pair as discussed above.

Using A's private key, A generates the signature for message M

Following Steps:

1. Select a unique and unpredictable integer k in the interval [1,n-1]
2. Compute $k_p = (x_1, y_1)$, where x_1 is an integer
3. Compute $r = x_1 \bmod n$; If $r = 0$, then go to step 1

4. Compute $h = H(M)$, where H is the Secure Hash Algorithm (SHA-1)
5. Compute $s = k^{-1}\{h + d_r\} \bmod n$; If $s = 0$, then go to step1
6. The signature of A for reminder vector and Quotient vector is the integer pair (r,s)

After this we get reminder vector signature and the Quotient vector signature. Then the Entity B decodes the reminder vector and Quotient vector using below decoding algorithm.

Decoding Algorithm:

1. Input : Bi-tuple <R,Q>, Δ value
2. Convert Q from Base 10 to Base B
3. Let $QB=(q_1,q_2,\ldots.q_n)$ be the representation in Base B
4. Interpret R as a vector of Base Δ number
   For $1 \le i \le n$
   $I=q_i \times \Delta + r_i$
   Where q_i the ith digit of QB,r_i the i_{th} element of R.
5. Mi=I-1(i)
6. $M=(m_1,m_2,\ldots.m_n)$

Then the entity B generates signature for reminder vector and Quotient vector and compares the entity A and entity B signature. If the signatures are equal, both the users are authenticated otherwise the user is unauthenticated.

IV. Results

For experimental analysis we implemented our proposed work with Java programming language, this hybrid approach of authentication and data confidentiality gives more security of data while transmission, points can be considered which satisfies the elliptic curve equation and key computation parameters can be computed as specified in above algorithm for authentication.

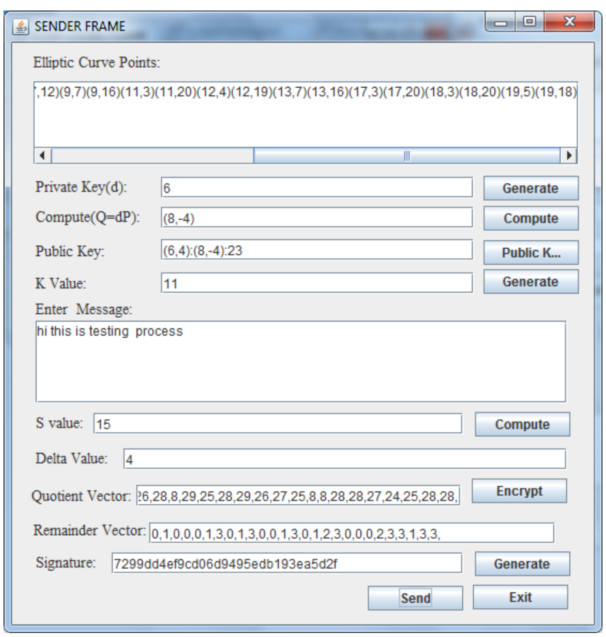


Figure2: Authentication process

V Conclusion and Future Work

In our proposed work we implemented a novel authentication method in a network using elliptic curve signatures with mod encoder. The confidentiality of the message is guaranteed by encoding one half, by and large Q,of the bittuple, the slip likelihood of disentangling is impressively high when either Q or R is obscure. The proposed method too gives a lossless pressure that encourages better transmission capacity usage and also as the encryption is connected to one portion of the encoded message, it reduces the computational complexity. It reduces the intrusion of the data remains the strong signature for the authentication.

We can improve our current research work with improved key management protocols for secure generation of key which is used to generate quotient and reminder vectors, we can improve the complexity of cipher by applying one more level encoding technique.

References

[1] G.R. Blakley, “Safeguarding Cryptographic Keys,” Proc. Am. Federation of Information Processing Soc. (AFIPS ’79) Nat’l 317, 1979.

[2] S. Berkovits, “How to Broadcast a Secret,” Proc. Eurocrypt ’91 Workshop Advances in Cryptology, pp. 536-541, 1991.

[3] R. Blom, “An Optimal Class of Symmetric Key Generation Systems,” Proc. Eurocrypt ’84 Workshop Advances in Cryptology, pp. 335-338, 1984.

[4] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, “Perfectly Secure Key Distribution for Dynamic Conferences,” Information and Computation, vol. 146, no. 1, pp. 1-23, Oct. 1998.

[5] C. Boyd, “On Key Agreement and Conference Key Agreement,” Proc. Second Australasian

[6] A. Shamir, “How to Share a Secret,” Comm. ACM, vol. 22, no. 11, pp. 612-613,1979.

Authors

First Author – V. Naga Malleswara rao, PG Scholar, Dept. of C.S.E. Prasad.V Potluri Siddhartha Institute of Technology , Vijayawada , India. Email:nagamalliv14@gmail.com.

Second Author – D.SreeLakshmi, M.Tech, Asst.Professor, Dept. of C.S.E., Prasad V.Potluri Siddhartha Institute of Technology, Vijayawada ,India.
Email:SreeLakshmidamineni@yahoo.com