ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

ElGamal Cryptosystem based group key management for Peer to Peer Communication

P.Vijayakumar[1], L.Jegatha Deborah[2]

[1,2] Department of Computer Science and Engineering, University College of Engineering Tindivanam, Melpakkam, India- 604 001

E-mail: vijibond2000@gmail.com, blessedjeny@gmail.com

Abstract

Designing a distributed group key management scheme for a peer to peer network with minimal computation complexity to support dynamic secure multicast communication is a challenging issue. In order to provide this facility, a new distributed group key management scheme is proposed in this paper for secure peer to peer communication. This proposed work is based on the existing ElGamal Cryptosystem . The main advantage of this proposed group key management scheme is that it minimizes the computation complexity of the peer user. The proposed algorithm has been implemented and analyzed with well known existing group key management schemes and observed that this proposed algorithm reduces the computation complexity significantly.

Index Terms—Computation Complexity, Peer to Peer Network, Storage Complexity

1. INTRODUCTION

Data exchange performed in group communication over a peer to peer network is insecure. A peer to peer (abbreviated to **P2P**) computer network is one in which each computer in the network can act as a client or server for the other computers in the network, allowing shared access to various resources such as files, peripherals, and sensors without the need for a central server. Peer to peer network is a distributed application architecture that partitions tasks or workloads among peers. Peers are equally privileged participants in the application. Each computer in the peer to peer network is referred to as a node. Peer to Peer networks can be classified as structured and unstructured. In Structured peer to peer networks, peers are organized following specific criteria and algorithms, which lead to overlays with specific topologies and properties. In unstructured peer to peer network, peers in the networks are connected in an ad-hoc fashion based on some loose set of rules. Unstructured peer to peer network is further classified into pure peer to peer system, centralized peer to peer system and hybrid peer to peer system. In pure peer to peer systems the entire network consists solely of equipotent peers. In the centralized peer-to-peer systems, a central server is used for indexing functions and to bootstrap the entire system. Hybrid peer-to-peer systems allow such infrastructure nodes to exist often called super nodes. Peer to Peer networks can be used for sharing files, or anything in digital format.

Various key distribution protocols have been proposed in the past for providing authentication and confidentiality. All the existing protocols include a set of key pairs, which include a public key that has to be broadcasted to a group of users and a private key which is kept as secret. The existing key management schemes are divided into two categories, namely centralized and distributed key management schemes. In centralized key management, a common Trusted Third Party (TTP) called a centralized administrator is used for generating and distributing the keys to group members. In contrast, the

ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online atwww.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

distributed key management scheme does not use a TTP. Instead, it allows each member of the group to generate and distribute the keys to other users. Each member of the group can derive a common group key from the key received from other group members. The key management protocol proposed in this project work is mainly based on the idea of distributed key management.

For distributed key management, one of the most common protocols is Diffie-Hellmen protocol which is developed in the year 1976. The main drawback of this approach is that, it is an asymmetric protocol and does not provide authentication to the data that has been exchanged between users. Hence, this approach may not be suitable for dynamic peer system to secure group communication. Therefore, in this paper there will be a pair of keys that includes a shared encryption public key and a private key. The shared or group encryption public key is derived from each user's public key who are existing in the group and the users private key remains secret.The cryptosystem used in our proposal is based on ElGamal cryptosystem. In this approach, both the private and public key has to be obtained, from which the group key is obtained. It is a computationally inexpensive cryptosystem and hence, it is suitable for dynamic peer systems, and for the secure group communication.

2. LITERATURE SURVEY

Diffie-Hellman proposed a one round two-party key agreement protocol [1] in which the users can exchange their data by using a secret key. The method is based on symmetric key agreement protocol, in which a common key is used for both the purpose of encryption and decryption. The main drawback of this approach is that it cannot be extended over group communication. Only a very few protocols have been published based on the extension of Diffie-Hellman to multi-party key agreement.

Xinliang et al(2007)proposed a Chinese Remainder Theorem based group key management protocol [2]. They introduced two protocols, namely Chinese Remainder Group Key (CRGK) and Fast Chinese Remaindering Group Key (FCRGK). In CRGK protocol to ensure security, the key server picks private keys from a pool of relatively prime positive integers. Since it deals with the dynamic network, forward and backward secrecy is ensured through FCRGK protocol. The main limitation of this project is that it is not suitable for Peer to Peer network since it requires a key server for the distribution of private keys to the users.

Only distributed protocols are able to work well in dynamic peer groups. Most of the existing protocols namely Joux protocol [3], BD protocol [4], ITW protocol [5] are inefficient, computationally expensive because they are not One round key agreement protocol and the members have to compute considerable modular exponentiations. All these protocols are symmetric, since a common secret key is used for both encryption and decryption.

The Joux protocol [3] build tripartite generalization of the Diffie-Hellman protocol by using weil and tate parings. These pairings were first used in cryptography as cryptanalytic tools to reduce the complexity of the discrete logarithm problem on some weak elliptic curves. The drawback of this method is that it is limited to only three participants and also the computational cost is higher.

Jeffrey Hoffstein et al(1998) proposed [6] "NTRU: A Ring Based Public Key Cryptosystem" which is an asymmetric key agreement protocol. In this method the message is encrypted by using the public key and it can be decrypted only by using user's private key. Since both

ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online atwww.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

encryption and decryption use only simple polynomial multiplication, it is computationally inexpensive. The limitation of NTRU is that sometimes decryption failure occurs for standard parameter. In such a case "chosen-cipher text attack" is possible.

3. PROPOSED WORK

An algorithm for asymmetric cryptography, invented in 1985 by Taher ElGamal, which is based on the difficulty of calculating discrete logarithms and can be used for providing confidentiality and digital signatures. The ElGamal Algorithm provides an alternative to the RSA for public key encryption. The main difference between RSA and ElGamal cryptosystem is that security of the RSA depends on the (presumed) difficulty of factoring large integer on the other hand, security of the ElGamal algorithm depends on the (presumed) difficulty of computing discrete logs in a large prime modulus. ElGamal has the disadvantage that the cipher text is twice as long as the plaintext. It has the advantage that the same plaintext gives a different cipher text each time it is encrypted.

3.1 Encryption / Decryption of ElGamal cryptosystem

Step 1: sender generates an efficient description of a multiplicative cyclic group G of order q with generator g. Sender chooses a random x from {0,….,q-1} Alice then computes $h=g^x$. Sender then publishes h, along with the description of G, q, g, as sender public key. Sender retains x as her private key which must be kept secret.

Step 2: To encrypt a message 'm', Bob chooses a random y from {0,….,q-1}. Receiver then calculates $c_1=g^x$. Receiver calculates the shared secret key $s=h^y$. Receiver convert his Secret message 'm' into an element of m' of G. Receiver calculates $c_2=m'.s$. Receiver sends the cipher text $(c_1,c_2)=(g^y,m'.h^y)=((g^y,m'.g^x)^y)$ to Sender.

Step 3: To decrypt a cipher text (c_1,c_2) with her private key x, Sender calculates the shared secret $s=c_1^x$. Then computes $m'=c_2.s^{-1}$ which she then converts back into the plaintext message 'm'. is inverse of s in the group G.The decryption algorithm produces the intended message, since

. = .( ) = . . = .

3.2 ElGamal Key Management Protocol

The sponsor user in the peer to peer network provides the public values p, b and to all the remaining users. The value of (i=1.....,n) must satisfy the requirement that mi = $2q_i$+ 1 and q_i(i =1, . . . , n) is an odd prime. The p value should be much smaller than q. The Initialization block generates the value and initializes the public values b and p. The sponsor user manages the key management module which includes Initialization block and Group key derivation block. The sponsor user is one of the users in the peer to peer network. Each user in the peer to peer network generates a private key sk_i from which respective public key h_i is computed. Each user sends their public key to the group key derivation block. The group key derivation block receives/collects public key of all the users in the network, which is managed by the sponsor user. The public key h_i and value is used to compute the group key H. This module computes the group key using Chinese Remainder Theorem. This block then broadcasts the computed group key to all the users in the network. This received group key is used for encryption of messages across the network. The user who wish to send the message will encrypt the message using the received group key. The encryption module broadcasts the encrypted message z_l and

ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online atwww.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

z_2 across the network. Decryption is done by decryption module. Each user uses values and their respective private key for decryption.

Step 1:Public parameter generation

The sponsor user in the peer to peer network generates the public values such as p, q, b and . The construction will be in *, where M is a product of n primes ….. . The value of (= 1, …,) must satisfy the requirement that = 2 + 1 (= 1, …. ,).The value = 2 + 1 (= 1, …. , and (= 1, … , is an odd prime.The value of p should be smaller than q. All these values will be shared among the group for the computation of group key.

Step 2: Group members contributions

The group key derivation block receives public keys of all the users who are existing in the network. Both the public keys and values of each user is used to compute the group encryption key. The group encryption key is derived based on Chinese Remainder Theorem (CRT) and the parameters received from group users

The public keys of user is computed by using the equation (1)

$h \equiv$ (1)

Where is the corresponding secret key of the user. The value of b represents the common primitive root modulo of each .

Step 3: Group encryption key derivation

The group encryption key 'H' is derived by using the following equation (2)

$\equiv \sum$ ( ) (2)

Here M = , … , = / and ≡ 1( ) .

Each member of the group can send any message to all the other group members by encrypting the message by using this group encryption key. Each user of the group can decrypt the cipher text encrypted by the group key 'H' using their own private key .

Step 4: Encryption and Decryption

Every user (= 1, . . ,) can encrypt the plaintext 'm' by using the group encryption key 'H'. The cipher text are given by the following equation (3) and (4),

≡ (3)

≡ (4)

Where 'r' is the random integer.

The user (= 1, . . ,) can decrypt the cipher text (,) using the secret key . The decryption formula is given by the equation (5),

≡ ( ) .(5)

Thus, the plaintext 'm' is obtained.

Considering the security, the smallest prime of q_i (i = 1, . . . , n), say q_j must satisfy the additional requirement that $q_j^{1/2}$ is sufficiently large. The security of this construction is based on the security of the ElGamal cryptosystem in Z^*_M . Its PFS property can also be achieved easily by multiplying public key H_i by an ephemeral random secret from Z^*_{mi} when it is broadcasted to other members.

MATHEMATICAL PROOF

$m \equiv z_2(z_2^{sk}{}_i)^{-1} \bmod m_i.$

$m \equiv mH^r \bmod M\{(b^r \bmod M)^{sk}\}^{-1} \bmod m_i$

$m \equiv m(\sum {}_i y_i (b^{sk}{}_i) \bmod M.)^r \bmod M\{(b^r \bmod M)^{sk}\}^{-1} \bmod m_i$

$m \equiv \{m (\sum {}_i y_i (b^{sk}{}_i))^r \{(b^r)^{sk}{}_i\}^{-1} \bmod M\} \bmod m_i.$

$m \equiv \{m (\sum {}_i y_i (b^{sk}{}_i))^r \{(b^r)^{sk}{}_i\}^{-1}\} \bmod m_i.$

ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

$m \equiv \{m (\sum {}_{i}y_i)^r (b^{sk}{}_i))^r \{(b^r)^{sk}{}_i\}^{-1}\} \bmod m_i.$

$m \equiv \{ m (\sum {}_{i}y_i)^r (b^{sk}{}_i{}^r) (b^{rsk}{}_i)^{-1} \} \bmod m_i.$

$m \equiv \{ m (\sum {}_{i}y_i)^r \} \bmod m_i.$

$m \equiv m\{ (\sum {}_{i}y_i)^r \bmod m_i.\}$

$m \equiv m\{ (\sum {}_{i}y_i) \bmod m_i.\}^r.$

$m \equiv m (1)^r.$

$m \equiv m.$

4. IMPLEMENATION RESULTS

This section shows the implementation results of our proposed work.

4.1 Initialization phase

Input: Number of users and bit size for key value.

Output: Generates the m value for n users, common primitive root b and p value.

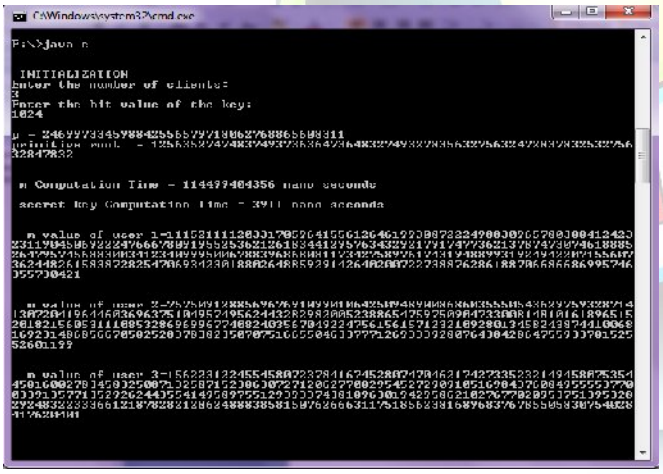

Fig.1 Basic parameters generation of ElGamal

Fig.1 shows the Initialization phase. For calculating the group encryption key, the sponsor user of the group computes and generates the value of for all the users of the group. The value of p and primitive root will be shared among the users.

4.2 Private key generation

Input : Number of users and key size.

Output: Generates secret key randomly.

Fig.2 Private key generation of ElGamal

Fig.2 shows the private key generation of the users. The private key generated is unique to ensure security. The private key generated will remain secret.

4.3 Public key generation

Input: secret key, common primitive root and m value.

Output: calculates the public key value

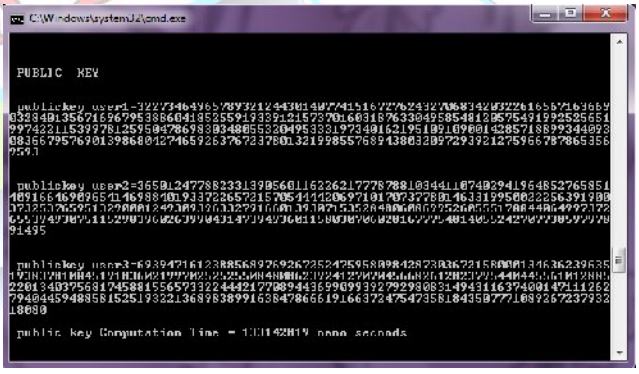

Fig.3 Public key generation of ElGamal

Fig.3 shows the generation of the public key of the user. The public key value generated is shared among the group. This key value is used for the computation of the group encryption key.

ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online atwww.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

4.4 Group key derivation

Input: Public key, x value, y value and M value.

Output: calculates the group key value.

Fig.4 shows the derivation of the group encryption key. The group encryption key is derived from each user public key values. The message to be sent is encrypted only by using the group encryption key.

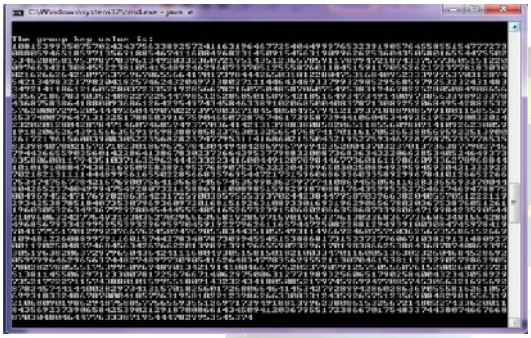

Fig.4 Group key derivation of ElGamal

4.5 Encryption

Input : message, r value, group key value and M value.

Output: calculates the z_1 and z_2 value.

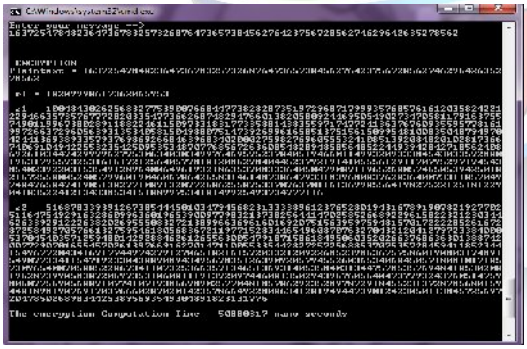

Fig.5 Encryption of ElGamal

Fig.5 shows the process of encryption. The message is encrypted by using the group encryption key. The encrypted messages will be transmitted to all the remaining users of the group.

4.6 Decryption

Input : $z_1$value, z_2 value ,secret key and m value.

Output : calculates the message.

Fig.6 Decryption of ElGamal

Fig.6 shows the decryption of the cipher text sent by the sender. The user can decrypt the cipher text by using their own private key which is kept as secret..

Table 1. Computational time complexity of various key management algorithms

The proposed method has been simulated in JAVA and we have analyzed the initialization of public parameters, key generation, group key derivation, encryption and decryption modules computation time with existing approaches. Table 1 shows the measured computation time in milliseconds for

	NTRU(ms)			*ElGamal(ms)*		
	256 bits	*512 bits*	*1024 bits*	*256 bits*	*512 bits*	*1024 bits*
Public parameters	*361*	*875*	*1154*	*351*	*687*	*1156*
Public key	*439*	*852*	*1081*	*26*	*31*	*48*
Group key	*133*	*230*	*561*	*17*	*31*	*48*
Overall computation time of sponsor user	*1372*	*2809*	*3877*	*491*	*5100*	*9163*
Encryption (computation time of sender)	*156*	*321*	*796*	*17*	*24*	*35*
Decryption (computation time of receiver)	*321*	*635*	*940*	*13*	*16*	*49*
Overall computation	*4567*	*10984*	*22330*	*3765*	*7802*	*19707*

such comparisons. It is evident from the values that the

ISSN2394-3777 (Print)
ISSN2394-3785 (Online)
Available online atwww.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 9, September 2015

computation time for our proposed algorithm is found to be better than the other algorithms.

5. CONCLUSION

The existing NTRU cryptosystem deals with polynomials for both encryption and decryption. Its security is based on the difficulty of analyzing the result of polynomial arithmetic modulo. This cryptosystem is computationally expensive due to larger public keys. ElGamal cryptosystem requires one exponentiation for encryption and two exponentiations for decryption. Hence this approach is less time consuming. Therefore, we have concluded that the proposed and implemented Elgamal cryptosystem is computationally inexpensive and it is secure against all the known attacks.

REFERENCES

1. W. Diffie, M. Hellman, New directions in cryptography, IEEE Transactions on information Theory ,22 (6), 1976, pp 644–654.

2. X. Zheng, C.T. Huang, M. Matthews, Chinese remainder theorem based group key management, in: Proc. of the 45th ACM Southeast Regional Conference, 2007.

3. A. Joux, A one round protocol for tripartite Diffie–Hellman, Journal of Cryptology 17 (2004) 263–276.

4. M. Burmester, Y.G. Desmedt, A secure and efficient conference key distribution system, in: A. De Santis (Ed.), EUROCRYPT 1994, in: LNCS, vol. 950, Springer,Heidelberg, 1995, pp. 275–286.

5. I. Ingemarson, D.T. Tang, C.K. Wong, A conference key distribution system, IEEE Transactions on Information Theory IT-28 (5) (1982) 714–720.

6. J. Hoffstein, J. Pipher, J.H. Silverman, NTRU, a ring-based public key cryptosystem, in: ANTS-III, in: LNCS, vol. 1423, Springer-Verlag, Berlin, Heidelberg, 1998, pp. 267–288.