Group Key Management Protocol: Secured Transmission in Compliant Groups

Amol Bhagat* and Lovely Mutneja**
* Innovation and Entrepreneurship Development Center,
Prof Ram Meghe College of Engineering and Management, Badnera, Amravati, India
amol.bhagat84@gmail.com
** Department of Computer Science and Engineering,
Prof Ram Meghe College of Engineering and Management, Badnera, Amravati, India
lmutneja@gmail.com

Abstract: Popularity of group-based application and related protocols leads to exponential growth of group communication. One form of group communication is Cooperative group. Cooperative group consist of multiple nodes/hosts connected together in resource sharing environment. Types of Cooperative groups are Wireless Mesh Network and Mobile ad hoc Network. The specified communication is totally unsecure. Secured transmission is one of the complex issues when dealing with the groups. A proper management of keys is needed in the form of generation, exchange, storage and replacement of keys i.e. a key management is needed for the management of cryptographic keys in a cryptosystem. Group Key Management Protocol (GKMP) provides an effective key management solution which deals with generation of keys and distributes them among different communication peers. This paper addresses the use of GKMP protocol for providing security of data transmission in cooperative group. The process of creating group, distributing keys to each member, controlling transmission, implementing security, and deleting group members is described in this paper. The comparison recent approaches are presented in this paper. This paper also describes the proposed methodology for group management. The performance evaluation of the proposed methodology is presented on the basis throughput, time required for key generation and time required for key distribution.

Keywords: Cooperative Groups, Group Key Management Protocol, Key Distribution, Key Generation, Secured Transmission.

Introduction

Group based applications and protocols have gained their popularity as they are used to provide efficient packet delivery from source to destination/receivers. Security is one of the main concern here as these involves communication over open networks, therefore proper key management is needed for providing security for a communication in groups. The main building block for achieving security in group communication scenarios is management of the secret information, this information is known only to the participants and therefore it is known as secret group key. In secure communication two entities should communicate without the intervention of a third party to listen in. Proper care must be taken so that eavesdropping or interception should not occur in such communication. In secure communication people can share information by certain means with varying degrees of certainty that third parties cannot intercept the data exchanged during communication.

The basic working of GKMP [1] is to provide group wise key in groups, support to dynamic joining and leaving in group, support to scalability, centralized operation. With the help of these features of GKMP protocol a foolproof solution for secure transmission can be designed. The basic purpose of GKMP is to deal with security related issues. It first creates keys for cryptographic groups for security. It also provides capability for distribution of keys, provides keys access control, known compromised host's access denial, and also provide controlling of group actions. The key generation methodology is cooperative generation which is implemented between two protocol entities. As the keys are generated then GKMP distributes the keys to verified GKMP entities. Verification means are provided for the means of distribution. GKMP implements review concept in which protocol entities passes permission certificates (PC), this process is perform during key distribution and is considered as a part of key distribution. The PC supported provides means of access control information, which leads to checking of permissions and comparing the level of service requested by GKMP entity. The service is denied if the permissions is less than or equal the request. These features of creating, distributing and providing means of access control information are the key concept for the proposed algorithm.

The proposed work with help of GKMP consist of compromise recovery feature in which GKMP distributes a list of compromised entities to group members during key management actions. The use of Compromise Recovery List (CRL)

allows members of groups to deal with compromised entities by implementing dropping of the connection action. GKMP consist of inbuilt feature to control group actions. In several situations while dealing with certain networks it is mandatory for a higher authority to control the generation of groups. Identification of group key controller, group key creation, group key distribution and group rekey [1] are the main objectives of the proposed approach in this paper.

Related Work

There are various approaches for providing security to data being transfer and for effective key management. Cryptosystem helps in sharing secret keys among users in an insecure channel e.g. satellite broadcasts, IP multicasting, TV subscription etc. Behzad Malek and Ali Miri [2] proposed secure broadcast encryption scheme with the use of short cipher texts. The algorithm generates fixed size cipher text. Leaving and joining of members in the group is quite easy, it does not affect any public parameters or private keys of existing members. This algorithm specifically works with small cipher text and provides greater security as compared to other models.

B Rong [3] proposes a pyramidal security model to deal with all these issue in which multi security level are maintained. This model consists of hierarchical security groups and multicast groups. For efficient key management solution in multicast groups, three schemes has been proposed Star key. Graph, separated tree key graph and integrated star key graph for key management in multicast group schemes used are: Diffie Hellman algorithm extended contributory key management, computational number theoretic approach, and logical key hierarchy (LKH).

Now a day's network applications are based upon group communication, therefore providing security and other necessary service is main concern. For addressing this, a novel solution has been proposed which addresses scalability problem in group/multicast key management [4]. Recommendation is to treat group as a triple (U, K, R) where U is a set of users, K represent set of keys, and R is a user-key relation. Representation of secure group is given by means of key graphs. Three strategies have been used for securely distributing rekey message. Rekeying strategies and join/leave protocols are implemented. These strategies have the advantage that they are scalable to large groups and provide frequent joins and leaves.

Broadcast channel deals with transmission of data from a source station to every node within the network. Examples are LAN, Packet radio network, etc. Need arises to send secret message to various users at a same time. Application provides such service is called secure broadcasting applications. This kind of applications is useful when dealing with document distribution, teleconferencing, etc. Guang-Huei Chiou and Wen-Tsuen Chen [5] proposed concept of locking using Chinese Remainder Theorem (CRT). Secure lock is used to lock the session key. This lock system has various advantages: single cipher text copy is sent, efficient deciphering and secret keys held by each user are minimized. This secure lock helps in solving the secure broadcasting problem.

Broadcast encryption [6] deals with secure data transmission over insecure channel with changing set of user in the group. The public key broadcast encryption scheme is designed for to provide encryption in way that the public key is stored in user's device, or it is provided to the receiver along with cipher text. Here two fully collusion-resistant broadcast encryption schemes has been proposed (PKBE public key broadcast encryption) which are used by the stateless receivers. This scheme assures that it is secure against numbers of colluders. Again this scheme is also scalable as transmission cost not depends on the number of users. The proposed schemes are referred as Decision modified Bilinear Diffie-Hellman. There is controlled number of users available beyond the specific set who sometime receives the multicast. This kind of relaxation leads to the development of f-redundant establishment key allocations [7]. The scheme provides guarantee that the receivers are not more than f times to intended recipients.

Increase in group-orientation applications and protocols leads to increase in group communication e.g. Multicasting, videoconferencing application etc. Security services are necessary to all this aspects for maintaining communication privacy and integrity. For dealing with this key agreement in dynamic peer groups has been designed. Dynamic Peer groups require both initial key agreement (IKA) and auxiliary key agreement (AKA) operations, such as member addition, deletion, group fusion etc. For this the concept of CLIQUES protocol [8] was introduced which offers complete key agreement services. A CLIQUE is based on Diffie-Hellman key exchange method. Versa Key architecture [9] provides group wide keys and scalability. This architecture is suited where there is dynamicity in group for joining and leaving. Key distribution time is very less in Versa Key architecture. These are the various features of Versa Key Architecture.

Attack-Resilient Security Architecture [10], called ARSA. The need for bilateral roaming agreement establishment is eliminated by the use of ARSA and it has real-time interactions between numerous WMN operators. By using ARSA, each end user is not bounded to any specific network operator. Key agreement between a user and a serving WMN domain and efficient mutual authentication is supported by ARSA. It is also resilient to a wide range of attacks. ARSA is a homeless solution i.e. not bound to specific WMN operator. It is designed to be resistant to various attacks against WMN access.

In [11] an algorithm called OFT (One-way Function trees) is presented, its main purpose is to established cryptographic keys in large groups having dynamicity. It is a centralized algorithm based on one-way function trees. It is a bottom-up algorithm, it approximately halves the number of bits needed to broadcast to group members for the purpose of rekeying when a member is added or evicted. This algorithm provides complete forward and backward security, which means newly joined

node cannot read previous message, and deleted members cannot read future message. In OFT members are allowed to contribute entropy to the group key. It has the capability that even working on Pentium II processor, it can handle groups up to 10 million members.

As technology get updated, rapid increase in multimedia application and data also get increased, internet also allows for wide distribution of digital media data. Duplicity also got increased on digital data. It leads to many threats as digital documents are now easy to copy and distribute. [12, 13] focuses on three algorithms to provide security to data based on nine factors for achieving efficiency, flexibility, and security. Author claims AES [22] is better than DES and 3DES, by comparing on factors such as key length, cipher type, block size, developed, cryptanalysis resistance, security, possibility key, possible ASCII printable character keys, time required to check all possible key. Session keys and transport session keys are the two types of keys chosen by Group Key Agreement (GKA) [14], Key transfer protocol rely on GKA for choosing those keys. Each entity shares secret key during registration use for the purpose of encrypting session keys. Author has proposed an authenticated key transfer protocol which is based on secret sharing scheme; GKA [14, 15] broadcasts group key information to all authorized members, unauthorized users cannot recover the group key.

Group key agreement [16, 17] protocol is used for the establishment of cryptographic keys dedicated to group participants; key is based on each member contribution, over a public network. This key works as a secure bridge between participants. The proposed protocol is unauthenticated and secure against passive adversaries only. Various methods are available for reducing transmission overheads in broadcast encryption, which are generally based on assigning one key per partition by implementing partitioning the users one way key-chains. One of the method adapts 'skipping' chains on partition which consist of 'p' revoked users and other methods works with cascade chains on partition with layer structure, here 'r' represents number of revoked users. Scheme one has some transmission overhead and the same transmission overhead with the Subset Difference (SD) is inherent to scheme two. Author has proposed a new broadcast encryption scheme [18] with same transmission overhead is same as that of SD scheme for small r and will become smaller than the SD with growing r. Scheme has small Transmission overhead (TO) if 'r' is not very small, scheme has same TO with SD when 'r' is very small, Smallest TO for all 'r' when both schemed are combined.

Important Quality of service aspect when dealing with large groups when concentrated on dynamic membership is the time cost performing key updates in events of member joins and leaves. Join-exit-tree (JET) [19] key management framework has been proposed by the author. Initially, special key tree topology for key updates with join and exit subtrees is introduced, then for determining the capacity of join and exit subtree optimization techniques are employed. Confidential and authenticated data transmission over an insecure network is one of an important goal of cryptographic research. An algorithm is considered to be secure if it cannot be broken with standard resources. For an efficient cryptosystem key distribution is also very important. This work focuses on comparison of various algorithms with TACIT Encryption Algorithm [20] by comparing parameters like key length, block size, type and features and also this work investigates HDL implementation of TACIT Encryption Algorithm. By highlighting capabilities of various algorithm author claims RSA is the most secure algorithm whereas DES is the least secure algorithm. Considering the speed factor TACIT Encryption Algorithm with Dual Port RAM is the fastest whereas RSA is slowest. Considering area TACIT Encryption Algorithm consumes more area.

Confidentiality is one of the important aspects of data which is to be consider, Data needs protection. Security is also needed for avoiding unauthorized access or modification of data. Fast changes in technologies leads to more and more multimedia data generation and transmission with possibility of data being edited, modified, deleted and duplicated other threats are also been applied. Therefore accuracy and sensitivity of information is a big security and privacy issue, for this a firm and robust solution is needed. [21] Cryptography is a technique useful to provide protection to user information. It can be achieve by means of implementing encryption and decryption.

Proposed Approach for Group Key Management in Compliant Groups

Group communication becomes one of the important technologies. As far as security is concern it is to be considered for securely transmission of important and private data in groups. Various techniques are available for secure transmissions in group all are having their own pros and cons. This motivated to design a proper, effective, cost efficient, easy to implement design for securely transmission of data. The proposed approaches for group key creation, group key distribution, group rekeying, member initiated joining and hostile member deletion are described in the following sections.

Group Keys Creation

In the proposed model first the command is transferred to group. The group controller (GC) listen this command. The future group keys are created with the coordination of the first member and the GC. The appropriate cryptographic properties must be possessed by the resulting key this is achieved creating group keys on two separate hosts. The information is exchanged between the controller and the group members to simplify a well ordered group creation, management. This information is utilized for distinctively identifying the GC identity, its permissions, authorization to create keys, the future group's permissions, the compromise list current state, and information management relating to the keys being created. Asymmetric signature methodology is utilized for protecting this information from counterfeit. To verify net wide parameters such as host

permissions, public keys are widely maintained. To verify locally generated information (e.g. peer identity) the public key is sent with the messages. Due to this there is no need to store the host's public key. The key creation process has following goals:

1. cooperatively generate a GTEK and GKEK,
2. to verify the identity of the key creation companion by verifying the messages signatures by the key creators,
3. share public keys,
4. by signing the GC identification, group identification, and group permissions, validate the GC
5. allow the first member, signed by the group commander to receive the GC identity, group member identities, group identity, group permissions, and group rekey interval for remotely managed grouped.

Distributing Group Keys to Members

The other group members must get the group keys before the group is fully operational. The purpose of other group member initialization is as follows:

1. cooperatively generate a session key encryption key (SKEK) for the transmission of the GTEK and GKEK from the GC,
2. allow each member to verify the identity of the controller and vice versa,
3. to create the group allow each member to verify the authorization of controllers,
4. send the key packet (KP) (consisting of the GTEK, GKEK), GC identity, group member identities, group identity, group permissions, and group rekey interval to the other members,

Group Rekey

Rekey is a two-step function that involves message exchange between the GC and a first member and other members. For group creation messages between the first member and GC must be exactly described. For the purpose of distributing the new GTEK and the new GKEK messages between the other members and the GC are utilized. These functions are

1. send the new GTEK and new GKEK to the other members,
2. allow each member to verify the identity of the controller,
3. to rekey the group, group identification, and GC identification allow each member to verify the authorization of controllers,
4. the other members should receive the GC identity, group member identities, group identity, group permissions, and group rekey interval.

Join Initiated By Member

Joins initiated by member to the group is supported by the GKMP. When the group initiator does not need to control group membership except to verify that all members of the group follow to some previously agreed protocols such type of service is most striking. A potential group member must request the key from the GC, unambiguously identify themselves, pass their permissions, and receive the keys before joining group operations. Several messages are pass between the joining member and the GC during this process. The purposes of these messages are as follows.

1. Request group join from controller
2. for the transmission of the group traffic encryption cooperatively generate a SKEK and GKEK from the GC,
3. allow each member to verify the identity of the controller and vice versa,
4. to create the group allow each member to verify the authorization of controllers,
5. the other members must receive the KP, GC identity, group member identities, group identity, group permissions, and group rekey interval to,

Cooperative and Hostile Member Deletion

There are two types of member deletion scenarios - cooperative and hostile. The cooperative deletion scenarios is the removal of a trusted group member for some management reason (i.e., reduce group size; prepare the member for a move). The hostile deletion usually results in a loss of secure state at the member's site (i.e., compromise, equipment breakage). The two scenarios present different challenges to the network. Minimization of network impact is paramount in the cooperative scenario.

In the case of a hostile deletion, the goal is to return to a secure operating state as fast as possible. In fact there is a trade-off. The compromised group can be eliminated as soon as the compromise is discovered, but this may cripple an important asset. So security concerns need to be balanced with operational concerns. The cooperative deletion function occurs between a trusted member and the GC. It results in a reliable deletion of the group key encryption and GTEKs at the deleted member. This deletion is intended to be an administrative function.

The essence of the issues involves a tradeoff between security susceptibility and operational stability. All traffic on the network is stopped if a member is found to be vulnerable, from a security point of view. The group may prefer to live with the security leak if group traffic is supporting a critical operation, instead of interrupting the group communication. To

restrict access of compromised members the proposed approach provides two mechanisms. First, a Certificate Revocation List (CRL) is created for utilizing it during the group creation process. This list will not allow a vulnerable member to be encompassed in a new group. Second, the proposed approach facilitates creation of another group without inclusion of the vulnerable member(s). The proposed approach does not dictate whether or not the group may continue to operate with a vulnerable member. The proposed approach uses a mechanism to remove a vulnerable member by key that member out. This involves switching group operations in newly created group, without the vulnerable member. A group delete message is multicasts to remove old group.

Proposed System Working

Group Head and Group Members are the two building blocks. The main responsibility of the group head is to form key and distribute this secret key to members for joining the group. The Group head can create any number of groups and can send a broadcast signal to all members of the particular group. After the secret key distribution is over the members authenticate themselves with the group header using the secret key. Once the authentication is done, the group is being form. Now the group head can perform data distribution to all members in the group. New member can join the group for this purpose the group head generates keys and perform re-keying. Group head can remove any of the members from group. Once the group gets form, server can send any kind of data to nodes. The complete working of the group head is as shown. The process gets start by forming a network, once the network gets form, group head get decided which will generate the secret key. Once the secret key get distributed, the members who acquire the key by any means can participate in group forming, for that they have to provide this secret key to the group head as an authentication means.

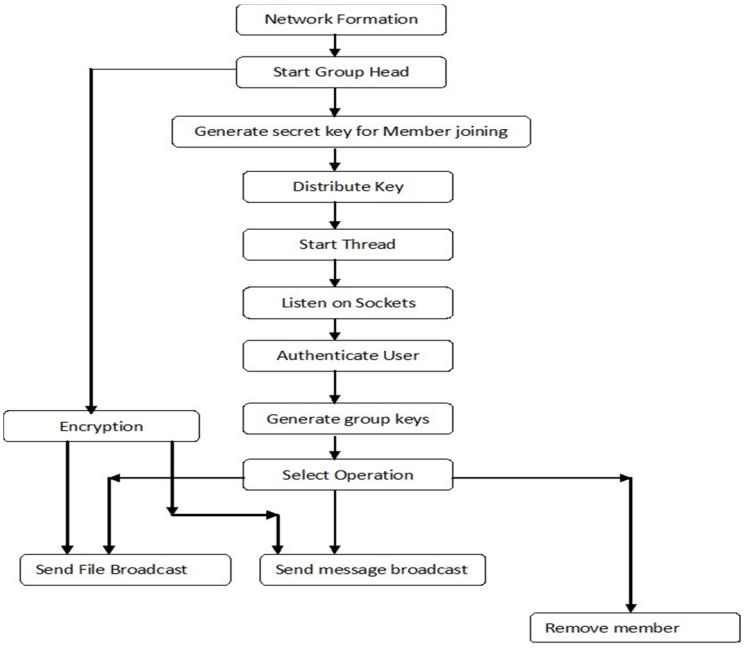

Fig. 1. Flow graph of proposed group head design

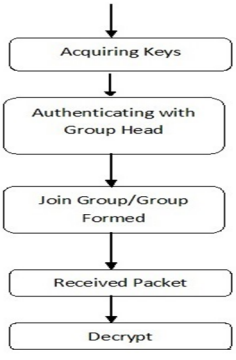

Fig. 2. Flow graph of proposed group member design

Group head observe the key and compares with its generated key if it matches it authenticate the member and joins in a particular group. Members in the network first acquire key from the head and then authenticate themselves to the group head to form group once the group gets formed they are able to receive the message in that particular group. The secret number from 1 to 99 as secret key is generated for joining of members to form a group. This method is linked with a Group Head class where group Head is used to generate numbers randomly as specified in code. Once the key been generated it is distributed to the other nodes for the purpose of group formation. The Group head authenticate with this particular key and listen on port number 19999 for all communication. After authenticating nodes a group is being formed with these authenticated nodes. Now the group head can perform any of the task i.e. can send a text message, can perform file transmission in group, or can remove the members from the group. All the data transmission either text or file transmission get performed with AES encryption and Decryption. Group Members acquires key for the purpose of group formation. This is the secret number generated by the group head and used for the purpose of authenticating the group members for group formation. Once the key being submitted the group head forms group. Group Head now can perform its task. The overall process is shown in figure 1 and figure 2.

Proposed System Implementation and Experimental Evaluation

The overall implementation of the proposed system is carried out in Java, on 2.8 GHz i3 processor, 4GB RAM and 3Mbps transmission rate. The Eclipse IDE is used for carrying out the implementation. The complete class hierarchy implemented in java is shown in figure 3. The main purpose of the cryptoClass is to provide the encryption and decryption mechanism. For providing this capability it consist of two methods encrypt() and decrypt(). It also consists of variable Plaintext and encryptionKey. The encrypt() is used in class GroupHead for the purpose of sending the message to the connected node. The decrypt() method is use for the retrieval of original message from the encrypted message. This decrypt() method is used in the GroupMember class for the decrypting purpose. The GKP class is used for the purpose of generating keys. These keys are distributed among different nodes. This class consist of various methods such as getCurrentKey(), setCurrrentkey(), getNextKey(), setNextKey(), setNextKey() and other methods. It consists of two main variables CurrentKey and Nextkey. The CurrentKey is a string variable which is used for the purpose for providing the Current key used. Next key is the generated key after the current key. First step is to identify the variable currentkey if it is less than 16 bit then padding is performed and if it is more than 16 bit then it is made 16 bit for operation.

The KeyGenerator class is a secret key generator used for the purpose of generating secret key used at the time of joining the group. It consist of generateKey() method which takes an integer argument to decide the length of the key. For generating the random key it used an object of a Random class. GroupHead class is a basic building block of this particular work. It consists of various data members as well as function. The first function is to start itself i.e. GroupHead(), it causes the Grouphead() to generate the secret keys for the purpose of group joining. Port 19999 is chosen to be a default port number for proposed work. It gets started on port number 19999 and is waiting to listen on this particular port number. Once it gets started it listens on port 19999, and will authenticate members according to secret key generated.

The GroupMember class first task is to provide the secret key to the group head for the purpose of forming groups. It first acquire secret key which is any secret number generated from the group head for the purpose to form group. This secret number is entered by the group member is used as an authentication by a group head. All broadcast are encrypted at group head and are decrypted at the group Members.

Proposed system works with the key length of 128 bit. Various performance metrics has been calculated and it has been identified 128 bit key is efficient from performance and security point of view. Key length has been decided by performing simulation in NS3 on different encryption algorithm and their key generation and distribution time has been calculated, throughput has been calculated by considering the total bandwidth of 3 Mbps as shown in table 1.

Table 1. Comparative analysis of key management and key distribution strategies

Key Size (bits)	Throughput (Mbps)	Time required for group key creation (seconds)	Time required for group key distribution (seconds)
64	2.916	0.523	1.352
128	2.892	0.793	1.788
256	2.809	1.493	2.480
512	2.769	2.456	2.982

Experimental result for Encryption algorithm AES, DES and RSA are shown in table 2, which shows the comparison of three algorithm AES, DES and RSA using same text file for five experiments i.e. 32 KB, 64 KB, 128 KB, 256 KB and 512 KB. The computational time is considered the time that an encryption algorithm takes to produces a cipher text from a plain text. It is used to calculate the throughput of an encryption scheme, and is calculated as the total plaintext in bytes encrypted

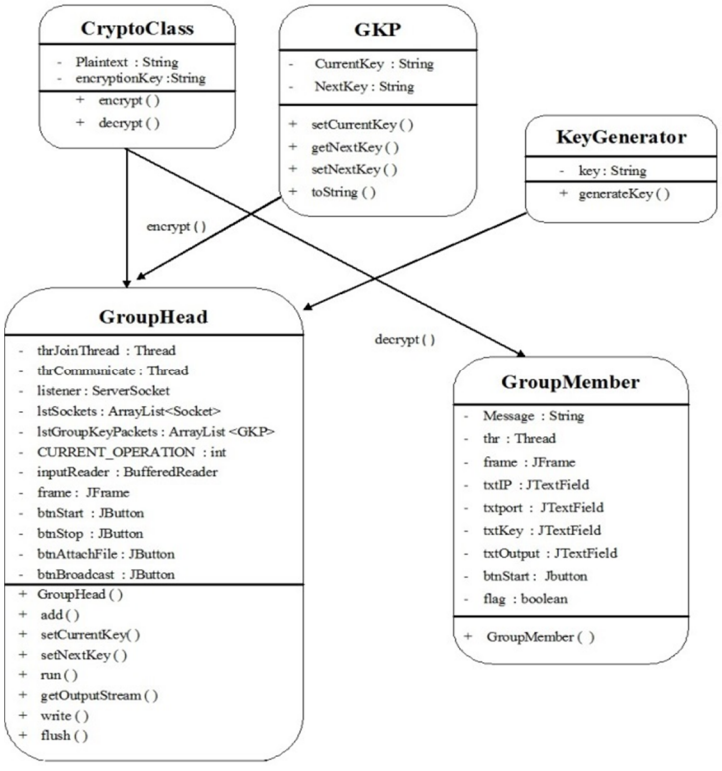

Fig. 3. Classes developed for carrying out experimentations using proposed approach

divided by the encryption or decryption time. It can be observed that time taken by RSA is very much higher as compare to DES and AES. DES is not as secure as compare to AES. Therefore AES is selected for carrying out the proposed approach.

Table 2. Comparative analysis of AES, DES and RSA using time required for encryption

Data in KB	Time Required for encryption (Seconds)		
	DES	AES	RSA
32	1.81	2.02	9.45
64	1.83	2.13	10.53
128	2.03	2.29	11.41
256	2.14	2.47	16.27
512	2.43	2.63	24.44

Conclusion and Future Scope

In this paper the existing literature on group communication and various security mechanism and key management concepts from previous literature is presented for formulating the proposed approach. Proposed work is based on GKMP features for group security. It provides Group management; i.e. creation of group, adding members, deletion of members. Proposed work addresses Key management concept. It provides security to data being transferred by means of implementing encryption and decryption techniques. Encryption and decryption in this work is done with the help of AES implementation which is consider to be a highly secured, and fast in operation.

Group communication is considered to be an important aspect in today's networking environment. Proposed work looks after security by means of implementing GKMP concepts involving group management, and key management, but still there are various aspects which need to be address in future. Proposed work addresses only few group communication features, complete full duplex mode is needed to be address in future. In future multiple group communication needed to address. AES is highly secured and effective, more effective and more reliable technology can be implemented. Key Distribution can be more enhanced. In Future work can be extended to form a Group, initiated by Group member.

References

[1] Mutneja, L. S. and Bhagat, A. P.: Secured Transmission in Cooperative Groups Using Group Key Management Protocol. In: IEEE International Conference on Communication and Network Technologies, pp. 139-144. IEEE Sivakasi, India (2014).

[2] Malek, B. and Miri A.: Adaptively Secure Broadcast Encryption with Short Ciphertexts: IJ Network Security, vol. 14, no. 8, pp. 71-79, (2012).
[3] Rong B., Chen H., Qian Y., Lu K., Hu R., and Guizani S.: A Pyramid Security Model for Large-scale Group-oriented Computing in Mobile ad-hoc: IEEE Transactions on Vehicular Technology, vol. 58, no. 1, pp. 36-47, (2009).
[4] Kei C., Gouda M., and Lam S.: Secure Group Communications using Key Graphs. IEEE/ACM Transactions on Networking, vol. 8, no. 1, pp. 16-30 (2000).
[5] Chiou G., and Chen W.: Secure Broadcasting using the Secure Lock. IEEE Transactions on Cyber Security, vol. 15, no. 8, (1989).
[6] Park J. H., Kim H. J., Sung M. H. and Lee D. H.: Public Key Broadcast Encryption Schemes with Shorter Transmissions. IEEE Transactions on Security, vol. 54, no. 3, (2008).
[7] Abdalla M., Shavitt Y., and Wool A.: Key Management for Restricted Multicast using Broadcast Encryption. IEEE/ACM Transactions on Networks, vol. 8, no. 4, (2000).
[8] Steiner M., Tsudik G., Waidnere M.: Key Agreement in Dynamic Peer Groups. IEEE Transactions on Networks, vol. 11, no. 8, (2000).
[9] Waldvogel M., Caronni G., Sun D., Weiler N., Plattner B.: The Versa-key Framework: Versatile Group Key Management. IEEE Journal Networks, vol. 17, no. 9, (1999).
[10] Zhang Y., and Fang Y.: ARSA: An Attack-resilient Security Architecture for Multi-hop Wireless Mesh Networks. IEEE Journal on Networks, vol. 24, no. 10, (2006).
[11] Sherman A. T., and McGrew D. A.: Key Establishment in Large Dynamic Groups using One-way Function Trees. IEEE Transactions Software Engineering, vol. 29, no. 5, pp.444-458, (2003).
[12] Hamdan A.: New Comparative Study between DES, 3DES and AES within Nine Factors. (2010).
[13] Harney H., and Muckenhirn C.: RFC 2093: Group key management protocol (GKMP) specification: Network Working Group, Website http://www.ietf.org/rfc/rfc 2093.txt (1997).
[14] Masood W., and Rasheed T. A Flexible Communication of Group Key Agreement.
[15] Lin Y.: A Group Key Management Protocol Based on Weight-Balanced 2-3 Tree for Wireless Sensor Networks. (2011).
[16] Luca V.: A novel batch-based group key management protocol applied to the Internet of Things. In: Ad Hoc Networks vol. 11, no. 8, pp. 2724-2737 (2013).
[17] Augot D.: An Efficient Group Key Agreement Protocol for Ad-hoc Networks. Sixth IEEE International Symposium on World of Wireless Mobile and Multimedia Networks, 2005.
[18] Cheon J. H.: Skipping, Cascade, and Combined Chain Schemes for Broadcast Encryption. IEEE Transactions on Information Theory, vol. 54, no. 11, pp. 5155-5171 (2008).
[19] Mao Y.: JET: Dynamic Join-Exit-Tree Amortization and Scheduling for Contributory Key Management. IEEE/ACM Transactions on Networking, vol. 14, no. 5, pp. 1128-1140, (2006).
[20] Kaur M.: Comparison of TACIT Encryption Algorithm with Various Encryption Algorithms. International Journal of Electronics and Computer Science Engineering, ISSN-2277-1956.
[21] Mathur M., and Kesarwani A.: Comparison between DES, 3DES, RC2, RC6, Blowfish And AES. Proceedings of National Conference on New Horizons, (2013).
[22] Juremi J.: Enhancing Advanced Encryption Standard S-Box Generation Based on Round Key. International Journal of Cyber-Security and Digital Forensics vol. 1, no. 3, pp. 183-188 (2012).