International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 4 Issue: 4
ISSN: 2454-4248
303 – 310

Group Key Rekeying Technique with Secure Data Encryption in MANETs

C. Shanmuganathan
Research Scholar, CSE
Manonmaniam Sundaranar University
Tirunelveli, India

Dr. P. Raviraj
Professor / Department of CSE
GSSS Institute of Engineering & Technology for Women
Mysore, India

Abstract—A Mobile Ad hoc Network (MANET) is a collection of autonomous nodes or mobile devices that can arrange themselves in various ways and operate without strict network administration. Ensuring security in mobile ad hoc network is a challenging issue and most of the applications in mobile ad hoc networks involve group-oriented communication. In Mobile ad-hoc network, each node treated as a terminal and also acts as an intermediate router. In this scenario, multi-hop occurs for communication in mobile ad hoc network. There may be a possibility of threats and malicious nodes in between source and destination. Providing the security in MANET is entirely different from the traditional wired network. In the present scenario, various applications of the mobile ad hoc network have been proposed and issues are solved by using the cryptographic techniques. Mostly cryptographic techniques are used to provide the security to MANETs. Cryptographic techniques will not be efficient security mechanism if the key management is weak. The purpose of key management is to provide secure procedures for handling keys in the cryptographic technique. The responsibilities of key management include key generation, key distribution, and key maintenance. Several key management schemes have been introduced for MANETs. The Group key management scheme is an efficient method for key management in MANET. In group key management scheme, rekeying is used whenever a new node joins or existing node leaves from the group. In this paper, we propose a periodic rekeying method (PRK) and analyze the performance of LKH rekeying techniques in a group key management schemes. The symmetric encryption techniques are analyzed with different parameters, such as Throughput and Energy consumption. Security and performance of rekeying protocols are analyzed through detailed study and simulation.

Keywords—*Mobile ad hoc networks, Group key, Periodic rekeying, Cryptography.*

*****

I. INTRODUCTION

Mobile ad hoc networks (MANETs) are known to have many security problems because of open medium, dynamic network topology, decentralized control, no centralized authority, lack of facilities in mobile devices and no clear rules for protection. Many mobile applications in MANET such as military, emergency response networks, M-commerce, online gaming, and combined work are based on the concept of group communications. While designing protocols for secure group communication systems in mobile ad hoc networks faces many technical difficulties and there are two ways of attack such as inside and outside attack. To deal with attacks from outside, one way is to use a symmetric key called the group key. The group key shared among all the users in a group. The group key will encrypt messages sent by a member (sender) in the group. Only group members (receiver) with the group key are able to decrypt the messages. Thus, the group key protects group communication information shared by authorized members. Since there is no fixed infrastructure support in MANETs, key management must be accomplished in a fully distributed manner. This creates additional processing and communication overheads whenever the group key is rekeyed because of a group member leave or joins frequently. Many mobile resources constrained such as bandwidth, memory size, battery life, and computational power affected the security. Group formation or partitioning affects with many factors like eavesdropping and security threats, unreliable communication, no fixed infrastructure, and frequent changes in network topology due to user mobility.

Instead of assigning the individual key for all users, a secret key is used for the entire group is called a group key [9]. When a new member joins a group, the group key is rekeyed immediately to ensure that the new member cannot decrypt old messages, this requirement is called backward secrecy [4]. When an existing member leaves the group, the group key is rekeyed immediately to ensure that future communications cannot be decrypted by the outside member; this requirement is called forward secrecy. An algorithm that deals with the generation, distribution, updating, and revocation of group keys is called a group key management protocol.

II. ISSUES IN MOBILE AD HOC NETWORKS

The major issues and challenges convinced by the mobile ad hoc networks the environment are as follows

Lack of Infrastructure

The Lack of Infrastructure is one of the major issues in mobile ad hoc networks. The absence of centralized management makes the detection of attacks a very difficult problem because it is not easy to monitor the traffic in a dynamic and large-scale mobile ad hoc network.

Dynamic Topology

Nodes are mobile and can be connected dynamically in a random manner. Links to the network vary timely and are based on the vicinity of one node to another node. In MANETs, nodes can leave and join the network at any time. Due to which the network topology changes frequently.

 ISSN: 2454-4248
303 – 310

Limited Power

Mobile ad hoc networks are composed of low powered devices. These devices have limited energy, bandwidth and computation power as well as low memory sizes.

Bandwidth optimization

Wireless links have significantly lower capacity than the wired links. The use of wireless links makes the network wide open to attacks such as eavesdropping and active interference.

Scalability

The scale of the mobile ad hoc network keeps on changing all the time. The scalability is a more challenging task in Mobile ad hoc networks due to frequent mobility request received from the nodes.

III. GROUP KEY MANAGEMENT

A. *Secure and Efficient Key Management*

In a Secure and Efficient Key Management (SEKM) is one of the decentralized asymmetric key management schemes. This scheme based on a virtual certificate authority (CA). By using CA trust model provides a secure procedure for communication in between nodes. The certificate authority (CA) is distributed the secret key to all the nodes in a network, which could be nodes with standard or good equipped. The major involvement of the scheme is that SEKM is designed to provide efficient certificate share updating among servers and to quick response to the certificate updating. The basic idea of SEKM is that server nodes forming the essential service group for efficient communication. Subset of the server nodes initiates the share updates in each round with valid certificates. A Ticket based schemes introduced for updating the certificates. By using the ticket based scheme recently added nodes to be isolated from share updating.

B. *Group Key Management*

In the multicast group communication, the sender sent the message once, it will be received by all the nodes in a group. The major problem in the group communication is providing security. In order to provide secure communication, we can restrict unauthorized access to a group by using encryption technique. Shared encryption key mechanism can be used for authentication in a group communication. The shared encryption key is called a group key.

Group key management can be classified into centralized, decentralized and distributed [4], [5] and [7]. A centralized scheme uses a centralized key controller for key management tasks including key generation, assignment, distribution, revocation, etc. If the centralized key server is compromised, then the entire system will be collapsed. So another scheme decentralized key management was introduced. In the decentralized scheme, groups are divided into subgroups hierarchically to spread out the workload of a central controller. But in the distributed scheme all members in the multicast group [8] responsible for generating and distributing encryption key for secure communications.

IV. GROUP KEY REKEYING

A single key is used for an entire group instead of assigning the individual key for each node. That is a set of nodes form a cluster and assigning a key to the cluster. Each group has a cluster head to update the key whenever nodes are left from the group or a new node joins the group. This process is called rekeying. There are two stages of the rekeying process is called forward and backward access control. Figure1 shows the rekeying process of a group key mechanism. In the forward access control, the system rekeys after an existing user leaves or removed from the system. The departed user will not be able to access further communication in a group. In backward access control, the system changes the group key (rekey) whenever a new user joins the group and the new user will not be able to access the past group communication messages.

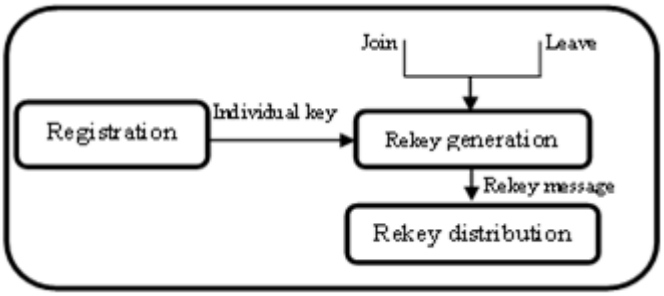

Figure. 1 Group key Rekeying Process

A. *Hierarchical Key Management Algorithms*

In a hierarchical group key Management scheme, group is formed as a logical tree structure. By using hierarchical tree, us can reduce the overhead incurred by the group members during the join or leave operation. The Group key controller creates a rooted balanced tree that has many leaf nodes as there are members. Each leaf node of the key tree is linked with a member of the group. Each internal node represents a logical subgroup. The root node represents the group key controller. The following mechanism explains about hierarchical key management scheme.

Logical Key Hierarchy (LKH): Logical Key Hierarchy provides a topological tree structure for handling the distribution of key while rekeying process in group key management. Figure 2 shows the LKH tree, leaf nodes are represented by the group members. The Group Key controller is responsible for distributing a secret key to the group members. Each member must have a group key and subgroup keys for its path to the root of the tree and a secret key. Assume that the member M6 will have group key, node E key, node B key and its personal key. If the member M6 were compromised, immediately the new group key distributes to the group members and also distribute the new keys for node E and B to

 ISSN: 2454-4248
303 – 310

the other members of subgroups. The node A key encrypting with group key would distribute to the key members M1, M2, M3, and M4. If a new member joins in the group, the group key controller needs to change the key to all the nodes in the path from a root to the members. Here the group key controller changes log n number of keys and encrypts two times i.e. 2 log n encryptions. Similarly, the leave request also takes 2 log n encryptions.

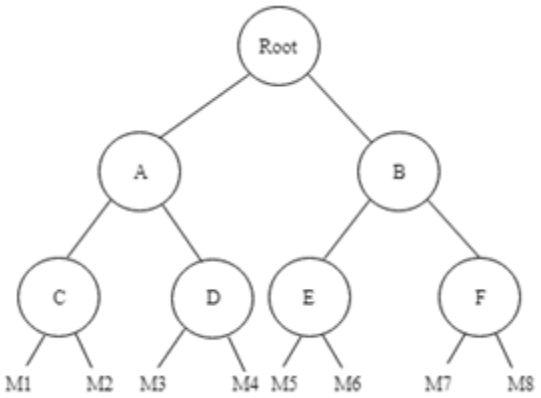

Figure 2 Logical Key Tree

B. Issues of Rekeying

There are two major issues in the individual rekeying process. First, it is very hard to control the synchronization problem while rekeying after every join and leaves operation. Due to the frequent mobility of nodes synchronization problem may occur. This will cause out of synchronization problem. The second major issue is communication overhead. Each and each joins or leaves request, the key will be rekeyed immediately in the individual rekeying process. The updated key shared to all the nodes if the key is changed. If the number of joins or leave request, then key sharing may also increase. This process will lead to communication overhead because most of the traffic is used for key sharing. Periodic Rekeying method (PRK).

In periodic rekeying, the nodes are not allowed to join or leave the network immediately. Whenever joining of the node and leaving the node requests is aggregated and rekeying is performed only in the specific interval of time. It reduces the rekeying process in dynamic group communications and to improve efficiency and reduce the problem that occurs. Individual rekeying is inefficient while compared to periodic rekeying in a dynamic and large network. The joining member has to wait until the next rekeying instance.

The periodic rekeying process can minimize the cost of the computation as well as the communication overhead. The performance of the periodic rekeying process is good for handling a large number of nodes in a group key mechanism. Re-keying operations of all members are synchronized to be carried out at the beginning of every re-keying interval. When a new member sends a join request, it also includes its individual blinded key.

The Figure 3 illustrates to join and leaves a request in the periodic rekeying method. Suppose M2, M5, and M7 nodes are leaving and a new node M8 wishes to join. The following steps to be followed: First node M8 broadcasts the join request, with its individual blinded key. Next, the leaf node 6 associated with M7 is replaced by the node M8, and then the leaf nodes 8 and 24 are removed. Nodes 7 and 23 are moved to node 3 and 11, respectively. Nodes M1, M4, M6, and M8 are nominated to be the sponsor nodes. M1 renews secret keys and K1, K0 and M4 renew K5, K2, and K0. M1 then broadcasts BK1 and M4 broadcasts BK5 and M8 broadcast BK2. M5 and M8 though having the sponsor role, do not need to broadcast any blinded keys as M4 has already broadcast this information.

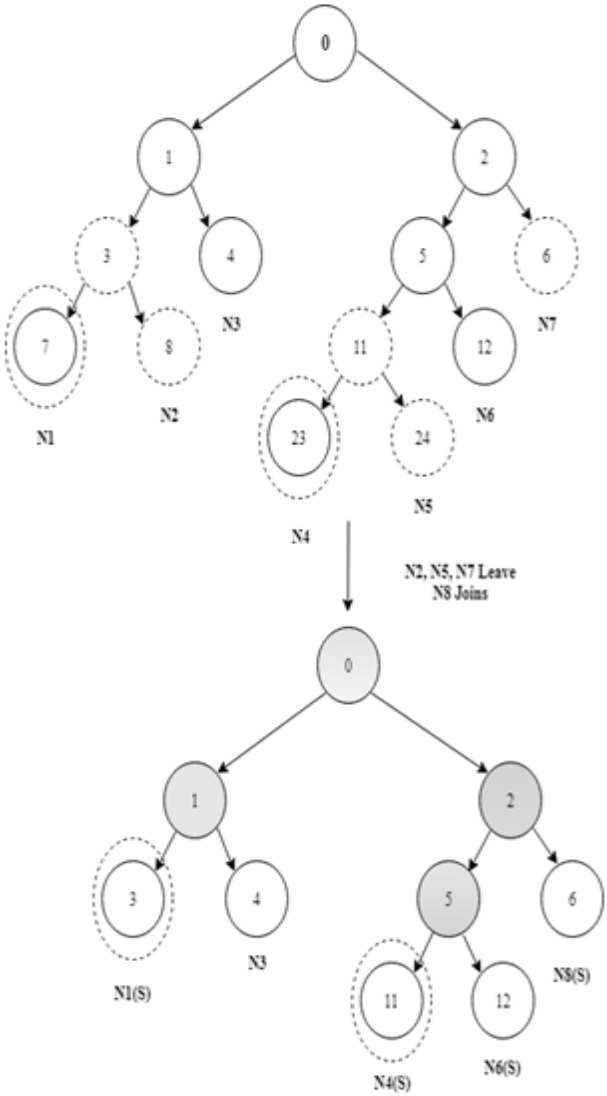

Figure 3 Group key PeriodicRekeying Process

V. SIMULATION RESULTS

The computation and communication cost of group key management schemes are done quantitatively and the results tabulated as follows. Using NS-2 simulation, their performance is tested and comparison results are showed in the graphs. In the simulation test, we have taken the experimental results up to200

ISSN: 2454-4248
303 – 310

nodes and the node size is increased 50 till 200, and corresponding values are plotted in the graph [29].

The simulation results show the variation of proposed method PRK with the LKH scheme. Figure 4 shows the communication cost of rekeying algorithms. We observe that the periodic rekeying algorithm has economical rekeying costs compared with LKH method. By reducing the number update messages between the hosts then we can reduce the rekeying cost. Departing members are selected randomly and joining members are then inserted into the key tree and the rekeying costs are calculated.

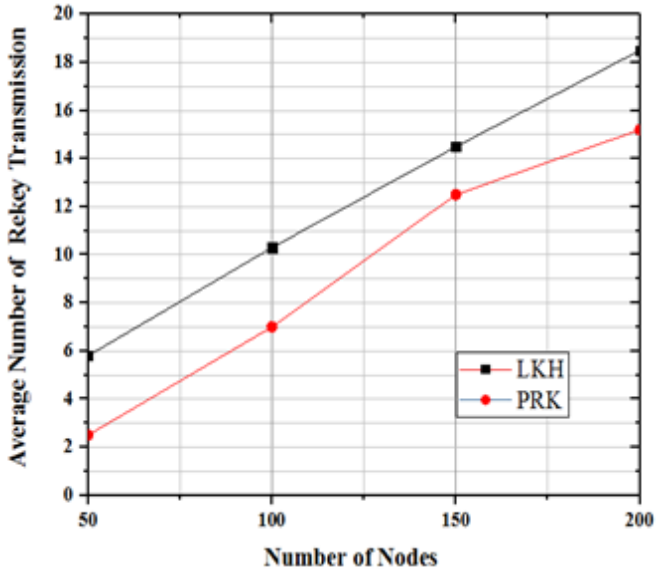

Figure4 Communication cost by varying number of nodes

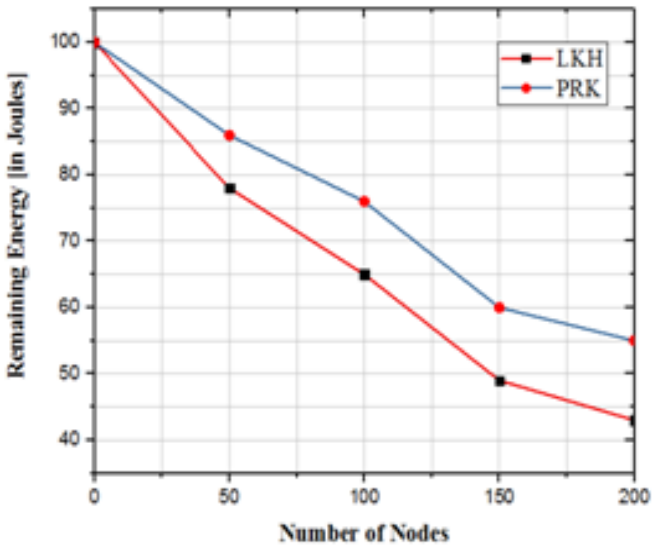

Figure 5 Energy Consumption by varying number of nodes

Figure 5 shows the energy consumption of rekeying algorithms. The graph clearly shows that periodic rekeying process has less energy consumption compared to LKH while increasing the number of nodes gradually.

Figure 6 shows that the periodic rekeying method achieves high throughput especially when a large number of joins or leaves request takes place in the specific interval of time.

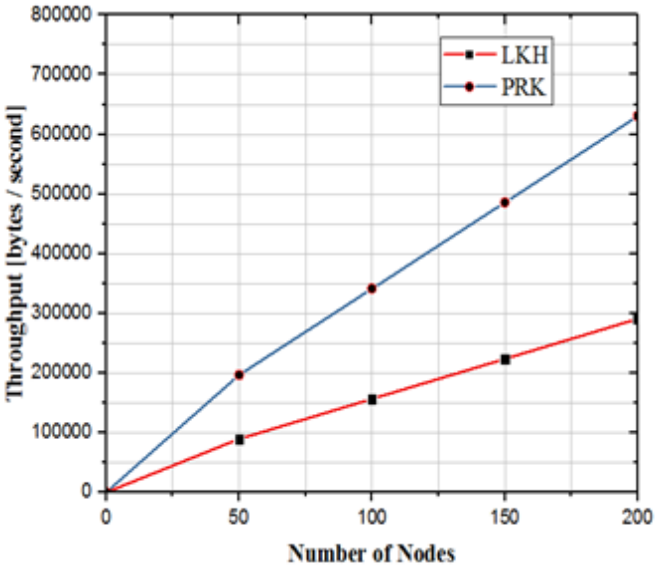

Figure 6 Throughputby varying number of nodes

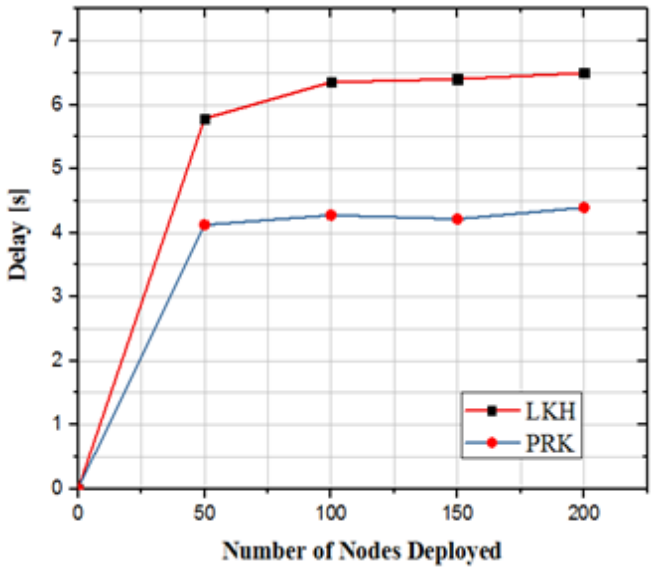

Figure 7 Average Delay by varying number of nodes

VI. PERFORMANCE ANALYSIS OF PRK

We analyze the performance of group key rekeying protocol in terms of communication cost, performance, security, and storage cost. The detailed analysis of the group key algorithms under the assumption that the key tree is completely balanced.

The mathematical analyses of the group key algorithms are based on two performance measures. First, the number of exponentiation operations gives a measure of the computation load of all members in the communication group. Second, the number of renewed nodes is said to be renewed if it is a non-leaf node and its associated keys are renewed. This metric measures the communication cost because the new blinded keys of the

 ISSN: 2454-4248
303 – 310

renewed nodes have to be broadcast to the whole group. For easy understanding, the following assumptions are made in the analysis: The existing key tree is completely balanced prior to the periodic rekeying event. Each member of a group has the same leave probability.

In the equation (1) let N be the number of members initially in the group, L be the number of new members who want to join the group. Let T denotes the existing tree and level of the node is l, and the maximum level of T is h. The key tree is initially balanced in the first assumption. Also, let R be the number of renewed nodes and E is the number of exponentiations. The performance measure which represents the number of exponentiation is composed of 2 parts namely exponentiation operations involved in computing the secret key which is done by all members and that of computing the blinded keys done by sponsors only. In a periodic rekeying algorithm, performance depends on membership leave position. Therefore, the expected number of renewed nodes is

$$\begin{cases} E[R\ batch\ ] = \{0, \quad if\ \ J = 0, L = N \\ \sum_{l=0}^{h-1} 2^1\ [\ 1 - (N - \frac{N}{2^1})] + (J - L)\ , otherwise \end{cases} \qquad (1)$$

According to the equation (1) if all the members leave the group and no members' joins, then the number of renewed nodes is zero.

A. *Communication cost*

The rekeying cost includes the total number of rekey messages that need to be sent to all authorized group members in a mobile ad hoc network in order for them to know the new group key. More bandwidth is needed for communication is called higher rekeying cost.

Last, the key storage denotes the number of keys each member need to store. The communication cost for secure group key distribution is determined by the numbers and types of logical paths that exist between two nodes in the network.

We now compare the communication cost of the periodic rekeying scheme (PRK) with that of LKH [6]. The reason for comparing our protocol with LKH is that it illustrates the differences in communication cost between protocols that were designed for a wired environment as opposed to a protocol that is geared towards wireless ad hoc networks. The periodic rekeying scheme for ad hoc networks in [2] shows that it is possible to reduce the cost of the original LKH scheme by 15% to 37% by mapping these physical locations of the members to the logical key tree in LKH for a static network. Because the performance overhead in [2] is of the same order as that of the original LKH scheme, we can also see the comparative performance of SEKM with respect to the protocol described in [2]. Note that we do not consider reliable group key [7] [14] delivery in the comparison, which actually biases the comparison in favor of LKH since it is a stateful protocol.

The metrics of interest are the average number of keys; a node transmits and receives respectively in every rekeying. We use the method of independent replications for our simulation. All the results have 95% confidence intervals that are within 5% of the reported values [20].

VII. SYMMETRIC KEY ENCRYPTION ALGORITHMS

A. *DES (Data Encryption Standard)*

Data Encryption Standard is a symmetric key block cipher encryption algorithm, which uses the Feistel ciphering structure shown in Figure 8. The input data (Plaintext) are divided into an equal sized block with 64-bit length. Pad bit is added to meet the required length in the last block. The input data length of DES is 64 bits, and the key length is 56 bit. The 64-bit input block is divided into two halves, left side 32 bit and the right side 32 bit.

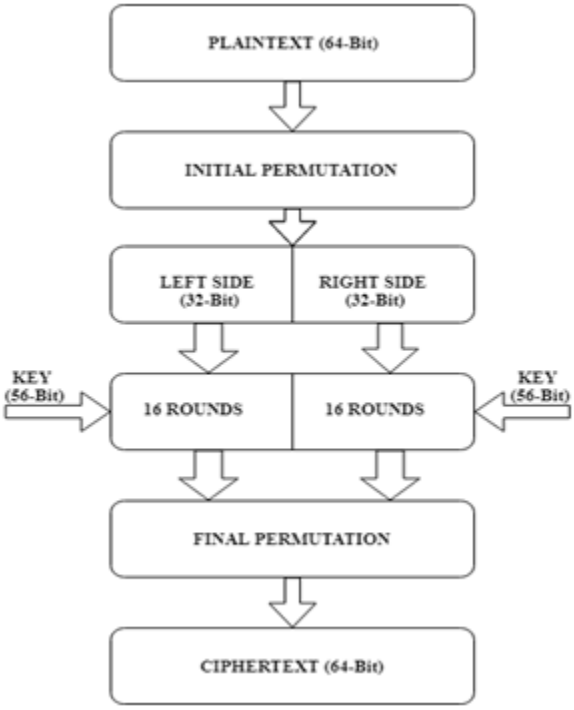

Figure 8 Overview of DES Algorithm

Each block of plaintext is encrypted using the private key into 64-bit Ciphertext by using permutation and substitution function. This procedure involves 16 rounds with different keys that are generated from the private key. The same steps will follow the Decryption process steps but in reverse order. Cryptanalysis has to find some weakness in DES algorithm if the key is weak.

Encryption C=E (K, P)
Decryption P=D (K, C)

 ISSN: 2454-4248
303 – 310

B. Triple DES Algorithm

Figure 9 depicts the Triple DES algorithm. This algorithm uses three stages of encryption and decryption process. Here first and the last stage uses the secret key K1 and the second stage uses the K2 key. This is much stronger than DES algorithm. Encryption and Decryption process of triple DES as follows.

Encryption C=E (K1, D (K2, E (K1, P)))

Decryption P=D (K1, E (K2, D (K1, C)))

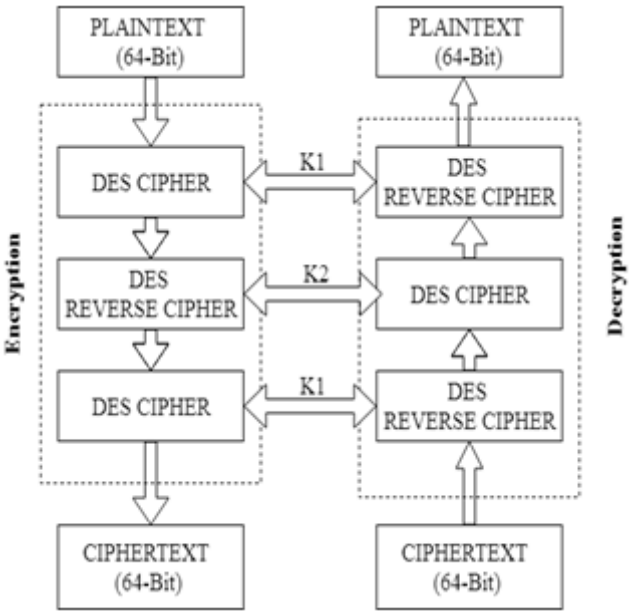

Figure 9 Overview Triple DES Algorithm

C. AES (Advanced Encryption Standard)

The more popular and widely used symmetric encryption algorithm is AES (Advanced Encryption Standard). In AES encryption algorithm accepts 128-bit input data, and the variable size key length (128 / 192 / 256). Unlike DES algorithm, the number of rounds in AES depends on the key size. In AES, 10 rounds, 12 rounds and 14 rounds use 128-bit, 192-bit, and 256-bit key respectively. Each round in AES consists of four processes; namely, subbytes, shiftrows, mixcolumn, and addroundkey. The decryption process of AES is similar to encryption process in reverse order.

VIII. Performance Analysis of Symmetric Key Encryption

A. Energy Consumption

Figure 10 depicts the differences in energy consumption by various encryption algorithms. The energy consumption was calculated using the mechanism described in [16]. The cost of encryption and decryption algorithm calculated based on Intel Pentium processor as clock cycles per byte shown in [17]. The Energy consumption calculated using the system with Pentium processor (Clock speed= 8980 cycles/sec). To calculate energy, use the following formula,

$$Energy = \frac{CCED}{PCS} * I * V$$

Where,

CCED = No. of clock cycles/byte during Encryption and Decryption

PCS = Processor clock speed in Cycles/sec

I = Total Current drawn (in Ampere)

V = Processor operating voltage (in Volts)

Energy = Energy in Joule

By using this formula, DES, 3DES, and AES algorithms consume 117.4, 280, and 42 Joules of energy run in 90, 216, and 32 clock cycles (per bytes) respectively. In term of energy, the AES consumes 64.4% and 85% less energy consumed by DES, 3DES algorithms respectively. Based on the Energy consumption, the AES algorithm is suitable for Mobile Ad hoc Networks because it consumes less energy than DES and 3DES algorithms.

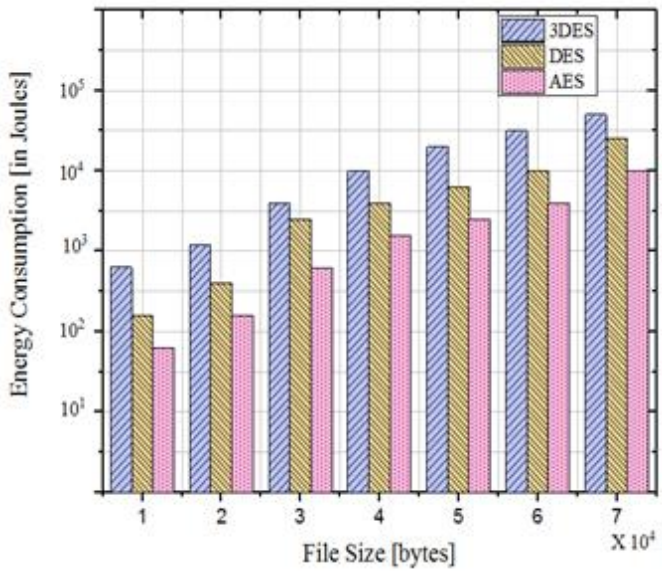

Figure 10 Energy Consumption for increasing file sizes

B. Throughput

Throughput calculation of the network while performing encryption and decryption operation is as follows,

$$Throughput = \frac{Input\ size}{Time\ duration}$$

Where,

Input size= Plain Text size in Bytes

Time Duration = Total time Consumed for Encryption and Decryption operation

By varying the file size, the throughput is calculated using the formula and the simulated results are plotted in the graph shown in Figure 11. The AES algorithm provides 34.1% and 47.6% better performance than DES and 3DES algorithms

 ISSN: 2454-4248
303 – 310

respectively. The analysis shows that the AES algorithm performs better than DES and 3DES algorithm. Hence, AES algorithm is suitable for Mobile ad hoc networks.

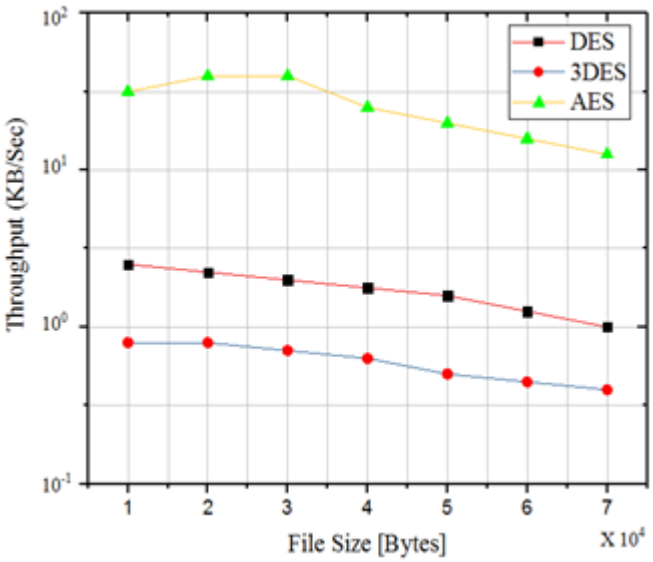

Figure 11 Throughput for different encryption algorithms

IX. Conclusions

In this paper, we analyzed and compared the performance of periodic rekeying algorithm. The periodic rekeying method provides a better performance than LKH method without sacrificing the security. Furthermore, reducing the number of decryptions can help to reduce the energy consumption, which, in turn, leads to battery saving. The periodic rekeying algorithm not only reduces the communication cost and also increases the overall performance of rekeying process. In periodic rekeying join events, the way the joining members are inserted in the key tree has significant effects, especially when there is a huge number of a request in a group. So the comparison results show that periodic rekeying method is suitable for a large group of nodes. The analysis shows that AES Encryption algorithm is suitable for encryption.

References

[1] Y. Kim, A. Perrig, G. Tsudik, Tree-based group key agreement, ACM Transactions on Information and System Security 7 (1) (2004) 60–96.

[2] L. Lazos and R. Poovendran. Energy-Aware Secure Multicast Communication in Ad-hoc Networks Using Geographic Location Information. In Proc. of IEEE ICASSP'03, Hong Kong, China, April, 2003.

[3] X. S. Li, Y. R. Yang, M. G. Gouda, and S. S. Lam. Batch ekeying for secure group communications. In Proceedings of Tenth International World Wide Web Conference (WWW10), Hong Kong, China, May 2001.

[4] S. Rafaeli, D. Hutchison, A survey of key management for secure group communication, ACM Computing Surveys (CSUR) 35 (3) (2003) 309–329.

[5] M. Younis, K. Ghumman, M. Eltoweissy, Location-aware combinatorial key management scheme for clustered sensor networks, IEEE Transactions on Parallel and Distributed Systems 17 (8) (2006) 865–882.

[6] C. Wong, M. Gouda,S. Lam. Secure Group Communication Using Key Graphs. In Proc. Of SIGCOMM'98, 1998.

[7] M. Eltoweissy, M. Moharrum,R. Mukkamala,Dynamic key Management in sensor networks, IEEE Communications Magazine 44 (4) (2006) 122–130.

[8] Y. R. Yang, X. S. Li, X. B. Zhang, and S. S. Lam. Reliable group rekeying: A performance analysis. Technical Report TR–01–21, The University of Texas at Austin, June 2001.

[9] C.K.Wong, M.Gouda, S.S. Lam, Secure group communications using key graphs, IEEE/ACM Transactions on Networking 8(1)(2000)16-30.

[10] Chang, I., R. Engel, D. Kandlur, D. Pendarakis and D. Saha, 1999. Key management for secure Internet multicast using Boolean function minimization technique. IEEE INFOCOMM, 2: 689-698.

[11] Ku, W.C. and S.M. Chen, 2003. An improved key management scheme for large dynamic groups using one-way function trees. Proceedings of the International Conference on Parallel Processing Workshops, Oct. 6-9, Kaohsiung, Taiwan, pp: 391-396.

[12] Parvatha, V.B. and S. Valli, 2005. EOFT: An enhanced one way function tree rekey protocol based on chinese remainder theorem. Proceedings of 20th International Symposium on Computer and Information Science ISCIS 05 Lecture Notes on Computer Science, 3733: 33-44

[13] Waldvogel, M., G. Caronni, D. Sun, N. Weiler and B. Plattner, 1999. The versakey framework: Versatile group key management. IEEE J. Selected Areas Commun., 17: 1614-1631.

[14] Alan, T.S. and D.A. Mcgrew, 2003. Key establishment in large dynamic groups using one-way function trees. IEEE Trans. Software Eng., 29: 444-458.

[15] Aldar C-F. Chan, Edward S. Rogers, "Distributed Symmetric Key Management for Mobile Ad hoc Networks", IEEE INFOCOM, vol.6,issue 1, feruary 2004.

[16] X.B. Zhang, S. Lam, D.Y. Lee, and Y.R. Yang, "Protocol Design for Scalable and Reliable Group Rekeying," IEEE/ACM Trans. Networking, vol. 11, pp. 908-922, Dec.2003.

[17] J. Pegueroles and F. Rico-Novella, "Balanced Batch LKH: New Proposal, Implementation and Performance Evalution," Proc.IEEE Symp. Computers and Comm. (ISCC), June 2003.

[18] S. Setia, S. Koussih, and S. Jajodia, "Kronos: A Scalable Group Rekeying Approach for Secure Multicast," Proc. IEEE Symp. Security and Privacy, 2000.

[19] C. Duma, N. Shahmehri, P. Lambrix, A hybrid key tree scheme for multicast to balance security and efficiency requirements, in: 12th Int'l Workshop Enabling Technologies: Infrastructure for Collaborative Enterprises, June 2003, pp. 208–213.

[20] P. Papadimitratos and Z.J. Haas. Secure Routing for Mobile Ad Hoc Networks. Proceedings of the SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002),2002

ISSN: 2454-4248
303 – 310

[21] Y. Hu, D. B. Johnson and A. Perrig. SEAD: Secure Efficient Distance Vector Routing in Mobile Wireless Ad-Hoc Networks Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'02), 2002.
[22] Yang Richard Yang, X. Steve Li, X. Brian Zhang, Simon S. Lam, Reliable Group Rekeying: A Performance Analysis, SIGCOMM'01, August 27-31, 2001.
[23] Menezes, A., Oorschot, P., and Vanstone, S. (1996). Handbook of Applied Cryptography, CRC Press.
[24] Hubaux, J., Buttyan, L., and Capkun, S. (2001). The Quest for Security in Mobile Ad Hoc Networks, In Proc. of the ACM Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc 2001).
[25] Sherman, T. and McGrew, A. (2003). Key Establishment in Large Dynamic Groups Using One-Way Function Trees. IEEE Transactions on Software Engineering, vol. 29, no. 5, pp. 444-458.
[26] Capkun, S., Buttya, L., and Hubaux, P. (2003). Self-Organized Public Key Management for Mobile Ad Hoc Networks, IEEE Trans. Mobile Computing, vol. 2, no. 1, pp. 52-64.
[27] Kim, Y., Perrig, A., and Tsudik, G (2000). Simple and Fault-Tolerant Key Agreement for Dynamic Collaborative Groups. In 7th ACM Conference on Computer and Communications Security, pp. 235244.ACM Press.
[28] Steiner, M., Tsudik, G., and Waidner, M.(2000). Cliques: A New Approach to Group Key Agreement. IEEE Transactions on Parallel and Distributed Systems.
[29] Rafaeli, S. and Hutchison, D. (2003). A Survey of Key Management for Secure Group Communication. ACM computing Surveys, vol. 35, no. 3, pp. 309-329.
[30] Burmester, M. and Desmedt, Y. (1994). A Secure and Efficient Conference Key Distribution system. In A. De Santis, editor, Advances in Cryptology – EUROCRYPT '94, no. 950.
[31] Wong, C., Gouda, M., and Lam, S. (1998). Secure Group Communications Using Key Graphs. In Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication, pp. 68–79.