A Pythagorean Tree based Key Generation Algorithm for Secure Group Communication in MANETs

B. Gopalakrishnan
Asst. Prof. Sl. Grade
Department of Computer Applications,
Bannari Amman Institute of Technology,
Sathyamangalam, India

A. Shanmugam, Ph.D
Principal
Bannari Amman Institute of Technology,
Sathyamangalam, Tamil Nadu, India

ABSTRACT
An efficient and secure group communication in Mobile Adhoc Network (MANETs) has brought attention to many security issues. In this paper, we propose Energy Efficient Dynamic Core based Multicast Routing Protocol to establish the path between the nodes that participate in group communication. During the routing process each node computes a Pythagorean Triple and constructs a Pythagorean Triple Tree (PTT) to generate a Contributory Key. Each node computes a pair of keys which will be exchanged among the nodes to compute group key for secure group communication. The rekeying operation is performed when the node join/leave the group. The proposed approach was analyzed on Computational cost, Group Formation, Percentage of Rekeying of nodes and Communicational cost with respect to Group Size. It reduces computational and communicational cost of the secure group communication when compared with other protocols.

General Terms
Key Generation Algorithm, Secure Group Communication

Keywords
Pythagorean Tree, Group Communication, MANETs, Group Key Generation Algorithm, Energy Efficient Multicast Routing Protocol.

1. INTRODUCTION
A MANET, a mobile ad hoc network, is an effective networking system facilitating an exchange data between mobile devices, without the support of wireless access points and base stations. A MANET is not restricted to unicast or multicast communication, but can also provide "many-to-many" transmission, which can be treated as a group communication, group communication has extensive applications, such as file and software updating, news feeds, video-audio transmission, multiparty video games, military application and dividing the works and cooperation in network.

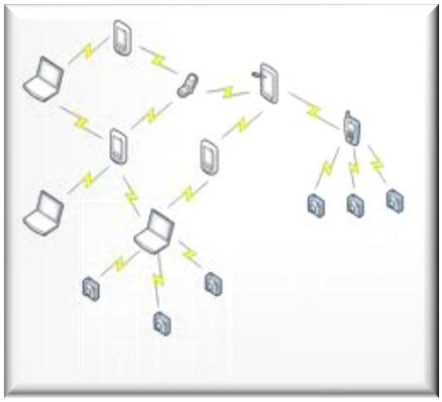

Fig 1. A model of Group Communication in MANETs

The group key management is the foundation stone of multicast security. It has been extensively studied in recent years. The group key management protocols are classified into centralized, decentralized and distributed group management. In centralized group key management protocols, there is a group key server (KS) which is responsible for group key distribution and updating. In decentralized group key management protocols, the groups are divided into several subgroups, there is a group key shared among all group members and every subgroup has a subgroup key among them. There is a group key server (GK) which serves for all members in groups and every subgroup has a subgroup key sever (SGK) which manages the subgroup key. Different from the former two, in the distributed group management protocols, many members in group are responsible for new group key generation and distribution. When a group key management protocol is proposed, the following factors should be considered: computation and storage requirement in key server, computation and storage requirement in all members, communication channel bandwidth requirement and rekeying latency. There are so many articles on group key management in these two decades and different research focus on different factors.

There are many routing protocols evolved to form group communication in Adhoc network like On Demand Multicast Routing Protocol (ODMRP), Dynamic Core based Multicast Routing Protocol (DCMRP) and Forwarding Group Multicast Protocol (FGMP), Core Assisted Mesh Protocol (CAMP) and Neighbor Supporting Adhoc Multicast Protocol(NSMP). In this paper the group is formed by Dynamic Core Based Multicast Routing Protocol. In this protocol the nodes are classified as Active, Passive, and Core Active nodes in the mesh network. Active and Core Active nodes are the nodes

that participate in the group communication whereas; passive nodes just forward the packets to other nodes. The primary advantage of DCMRP is performance improvement with respect to packet delivery ratio.

Pythagorean Triple

M. Beattie and C. Weather [2] have described a method for finding integer points on Hk for various k based on "growing the tree of primitive Pythagorean triples from (3, 4, 5)".

A point P = (a, b, c) € H0 satisfies $a^2 + b^2 = c^2$. If a, b, c are positive, then (a, b, c) is called a Pythagorean triple. Every Pythagorean triple is a multiple of a primitive Pythagorean triple, i.e. one in which gcd (P) = gcd (a, b, c) = 1.

In order to construct all primitive Pythagorean triples (a, b, c) with a, c odd and b even are obtained by switching a, b. The tree of primitive Pythagorean triples (a, b, c) with a, c odd and b even is constructed to the triple (3, 4, 5), then to (3, 4, 5) = (5, 12, 13), etc, so that each branching of the tree has 3 limbs. The first few branches of the tree are given in Fig 2:

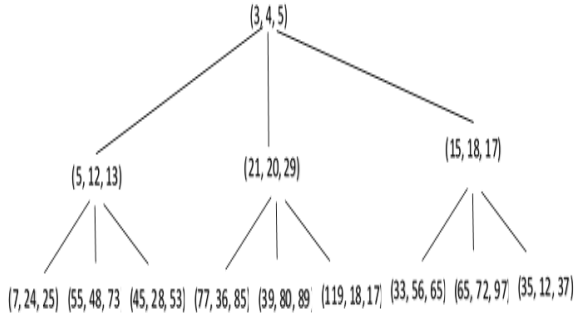

Fig 2. A Primitive Pythagorean Triple Tree

2. RELATED WORKS

Patrick P. C. Lee, et al. [9] proposed the Authenticated Tree-Based Group Diffie–Hellman (A-TGDH) protocol that provides key authentication for interval-based algorithms. Each member holds two types of keys: short-term secret and blinded keys as well as long-term private and public keys. They developed three interval-based distributed rekeying algorithms, termed the Rebuild algorithm, the Batch algorithm, and the Queue-batch algorithm.

Xuanwu Zhou, Ping Wei [11] analysed the security threats and system flaws of present key management schemes by combining (t, n) threshold cryptography and key management and presented a threshold key management scheme based on ECC (Elliptic Curve Cryptosystem). In the scheme, CA and KDC is a group composed of n members that share the CA/ KDC private key that is divided into n secret pieces. Only number less than t members can cooperate to recover the secret key to generate and distribute public key certificates or generate symmetric keys. Nitesh Saxena and Jeong Hyun Yi [7] proposed a technique which uses secret sharing based on bivariate polynomials. This scheme allows nodes in a MANET to readily and efficiently share pairwise secret keys without any centralized support. They found that B-BLS is more efficient than previous mechanisms, based on uni-variate polynomial secret sharing and threshold BLS signature, in terms of computation, communication, and energy consumption. Chu-Hsing Lin and Chen-Yu Lee [3] proposed the modified AKM to reduce the communication cost/computation cost to 1/t of the original cost without security loss. The comparison of the modified AKM with the AKM hierarchy suggests that modified AKM is more practical because it can handle huge numbers of dynamic nodes in MANET and provide sufficient security requirements.

Zhang Chuanrong and Liu Weijiang [13] proposed an ID-based signcryption scheme from bilinear pairings on elliptic curves. The proposed scheme is employed to design distributed key management with light-weight cryptographic technique. The proposed ID-based signcryption scheme is efficiency in terms of both the communication overhead and the computational requirement. It improves the security and efficiency of the previous key management protocols in MANET because of the application of the newly proposed signcryption scheme. AnahitaMortazavi [1] proposed an efficient many-to-many group key management protocol in distributed group communication in which the group members are managed in the hierarchical manner logically. When the node joining no keys to be exchanged between existing members, but on leaving one key, the group key is delivered to the remaining members.

Lu Li, et al. [4] presented a novel distributed key management scheme, a combination of certificate less public key cryptography (CL-PKC) and threshold cryptography, which eliminates the need for CL-PKC distribution, key distribution problem and also prevents single point of failure. It is a model for the use of public key cryptography that is intermediate between traditional PKI and identity-based public key cryptography (ID-PKC).ID-PKC enables a trusted Private Key Generator (PKG), which generates the private keys of the entities using their public keys and master secret key so it reduces the computation and improves the efficiency.

Yi-Ruei, et al. [12] proposed a group key management scheme based on a Meta Proxy Re-encryption (PRE) scheme. The first RSA-based PRE scheme for group key management and has the desired properties of uni-directionality and multi-hop. Their feature provides a practical solution for group key update when members go offline from time to time. It also improves the LKH method in many aspects, such as the number of secret auxiliary keys held by each member, the way of handling off-line members, etc.

JikaiTeng and Chuankun [5] proposed a security model for a certificate less group key agreement protocol and a constant-round group key agreement protocol based on CL-PKC. It does not involve any signature scheme, which increases the efficiency of the protocol. It formally proved that the protocol provides strong AKE-security and tolerates up to n−2 malicious insiders for weak MA-security. The protocol also resists key control attack under a weak corruption model. Min-Ho Park, et al. [6] proposed a new GKM scheme for multiple multicast groups, called the master-key-encryption-based multiple group key management (MKE-MGKM) scheme. This scheme exploits asymmetric keys a master key and multiple slave keys, which are generated from the proposed master key encryption (MKE) algorithm and is used for efficient distribution of the group key. Compared with other schemes, it is much simpler and can significantly reduce the storage and COs in the rekeying process, with acceptable computational overhead.

3. PROPOSED SYSTEM

In the proposed system a mesh topology is considered where nodes placed scattered in the network. The nodes want to communicate data or information in the mesh topology with each other in a secure and efficient way. To accomplish this task the group is formed by using DCMP routing algorithm where nodes are classified as Core Active Member (CAM), Active Member (AM) and Passive Member (PM) in the network and groups are clustered according to the routing path of the network. An efficient algorithm has been proposed to construct a group key by using Pythagorean Triple and each node will be assigned one value from the Pythagorean Triple and Pythagorean Triple Tree is constructed according. To maintain the group a joining and leaving algorithm is designed, that generates a contributory Group key accomplish secure communication among the group members.

3.1 Initialization

3.1.1 Node Creation

Create nodes in the mesh topology with the following data structure.

Table1. Node Structure

Notations	Description	Type
N	Node Identification	Integer
BL	Battery Level	Integer in percentage
NT	Node Type	CAM,AM,PM
PATH	Path list	Integer Array
RPATH	Reply Path list	Integer Array
CK	Contributory Key	Integer
a,b	Key Pair	Integer
GK	Group Key	Integer
Pkey	PrivateKey	Integer
Status	Node Status	String (ACK/NACK)
Maxhopcount	Maximum Hop Count	Integer
Hcount	Hop Count	Integer

3.1.2 Assigning the Node Type (NT)

Pseudo code for assigning Node Type

```
Assumptions
// Let n be number of nodes in the network and BL be battery level in
Percentage of the nodes in the network.
// Let N be an Array of Node Structure.
Node Type (Node N, Battery Level BL)
Begin
For i = 1 to n
Begin
If (N[i]. BL < 30) then
N[i].NT = 'PM';
Elseif (30 < N [i]. BL> 90) then
N[i].NT = 'AM';
Else
N[i].NT = 'CAM';
End
End
```

3.2 Establishing the Group

In this paper we assume the nodes are arranged in mesh topology and the nodes are classified into three different categories as Core Active Member, Active Member and Passive Member using Node Type algorithm as mentioned above.

Node Type (Node N, Battery Level BL).

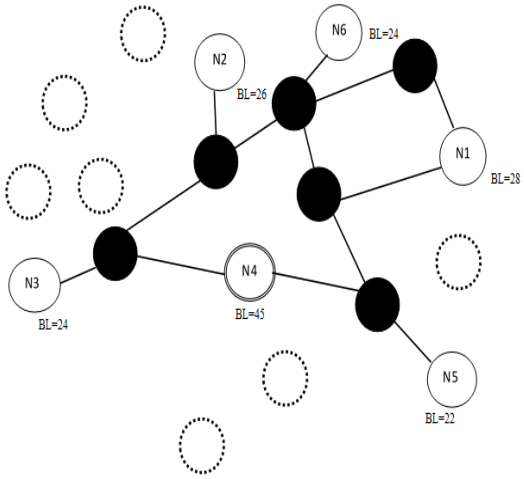

Fig. 3 Grouping nodes through DCMP

The following steps are proposed to establishment the group during Route Discovery phase of DCMP routing protocol.

a. Each Core Active Member and Active Member in the network will create the JoinReq message and flood the message to all one hop neighbours in the network.

JoinReq(SNid, PATH,Hcount,Maxhopcount);

b. Upon receiving the JoinReq message the core active members and active members will forward the message to next one hop neighbour in the network.

c. During each forward of JoinReq message the Node checks whether the Node Identifier (Nid) already not present in the PATH list then append the PATH list and Hcount = Hcount + 1. If the Hcount reaches the Maxhopcount value then its stops forwarding the JoinReq message in to the network.

d. The receiving nodes will set the STATUS as ACK or NACK send ReplyACK message to the source node that initiated the JoinReq message along the reverse path.

ReplyACK(Nid,Sid,RPATH,Status);

e. Each node will append the Node ID (Nid) of node in RPATH list during the ReplyACK message toward the source Sid. The RPATH list is used to generate the Group Key using Pythagorean Triple and construct the Pythagorean Triple Tree. The RPATH list is passed as parameter in Group Key Generation algorithm.

Pseudo code for Group Formation

```
//For each node in the mesh topology
Group Formation(n:  number of nodes in mesh network)
For i = 1 to n
Begin
 If N[i].NT == 'CAM' or N[i].NT == 'AM' then
```

Create a JoinReq(SNid,PATH,Hcount,Maxhopcount);
If (Hcount<Maxhopcount)
Begin
Forward JoinReq (SNid,PATH,Hcount,Maxhopcount) to all one hop Neighbors;
Hcount = Hcount + 1;
If N[i].Nid not in PATH then
Append PATH with N[i]. Nid;
End
Else Stopforwarding the JoinReq(SNid,PATH,Hcount,Maxhopcount) message;
End
Create ReplyACK (Nid,RPATH,Status) message
Forward in Reverse Path of the PATH list;
For I = n to 1
Begin
i = last node in the PATH list;
if (N[i].Status == ACK) then
Begin
Store the N[i].Nid in RPATH (Reply Path);
ForwardReplyACK (Nid,RPATH,Status) to next N[i];
Else
ForwardReplyACK (Nid,RPATH,Status) to next N[i];
End
End
Group Key Generation (RPATH);
End

3.3 Group Key Generation Algorithm

The following steps are performed to generate the group key at each node Ni.

a. Each Node in the RPATH is taken to generate the Contributory Key (CK) in the group.
b. The pair of nodes Ni and Nj are passed as parameter to the Pythagorean Triple Tree (Ni, Nj) to compute Contributory Key (CK).
c. The Highest Contributory Key is taken for Group Key Generation of the Node.
d. In Pythagorean Triple Tree (Ni, Nj) we compute Pythagorean Triple (a, b, c) that satisfies the Pythagoras Theorem ($a^2 + b^2 = c^2$) . Assign the Ni..Pkey = 'a' and Nj.PKey = 'b' return the value of 'c' to the group key generation algorithm.
e. Each node Compute N[i].a = CK / N[i].Pkey; N[i].b = K mod N[i].Pkey;

The pair of keys (a,b) will be exchanged among the nodes i and j to compute Group Key at each node i and j as

N[i].GK = N[j].a * N[i].Pkey + N[j].b

N[j].GK = N[i].a * N[j].Pkey + N[i].b

Pseudo code for Group Key Generation
Group Key Generation (RPATH List);
Begin
//For each node in the RPATH
// K an array to store intermediate Contributory Key;
i = 1 and j = 1
While (i< n)
Begin
K[j] = Pythagorean Triple Tree (N[i], N[i+1]);
i = i +2;
j = j +1;
End
//Find the Highest K value ie Contributory Key CK
For i = 1 to j
Begin
If (K[i] < K[i+1]) then
CK = K[i+1]
Else
CK = K[i];
End
Assign the Contributory Key to Every Node in the Group
For each node in the group will compute
For i = 1 to n
Begin
N[i].a = CK / N[i].Pkey;
N[i].b = CK mod N[i].Pkey;
End
If (Node i wants to communicate data to j) then
A pair of keys as ai,bi and aj ,bj are exchanged among i and j respectively;
Begin
Compute group key for node i
N[i].Gk = N[j].a * N[i].Pkey + N[j].b;
Compute group key for node j
N[j].Gk = N[i].a * N[j].Pkey + N[i].b;
End
End

3.4 Group Maintenance Phase

During the maintenance phase of the DCMP routing protocol the nodes may join or leave the node at any time in the Mesh topology Due to mobility the nodes will move from one location to another location and dynamic nature of the group nodes the nodes may join or leave the group at any time.

3.4.1 A Node Joining the Group

In this scenario we need to check whether the new node joins the mesh network or the non participating node wants to join the group.

The following steps are performed when a new node 'New' joins the group in the Mesh Network.

Algorithm:

The New node is passed for the Node Type (New id, Battery Level BL).

If the New.NT == 'CAM' or New.NT == 'AM' then

The new node performs Group Formation (New Node n);

That internally invokes the Group Key Generation Algorithm and computes a pair of keys (a,b) for that node.

Else

The New node search for Core Active Member node in one Hop neighbor in the Network.

Now the new node can communicate with any other node in the group by exchanging the pair of key (a,b)to compute the Group Key for that node.

The Encryption and Decryption is performed using the Group Key to exchange data/ information securely among the nodes.

3.4.2 A Node leaving the group

During the leaving process of the node it verify whether the node is CAM or AM member then it Leave Algorithm otherwise it just delete the information from the CAM member of that node. The leave operation is performed as follows

Algorithm

Step 1. The Node wants to leave the group will send leave request message (LeaveReq) to all the nodes in the group to get acknowledgement from every node that participated in the group communication.

LeaveReq(Nid , Path , Type, n)

Step 2. Upon receiving the LeaveReq message every node will send ReplyACK to the source node that sends the request to leave the group. If all nodes send ACK (Positive Acknowledgement).

ReplyACK(Nid,RPATH,Status)

During ReplyACK it stores the Nid in PATH list that used to remove the information from the others in the group.

The Group Key Generation algorithm is performed with the remaining nodes in the Group.

4. SIMULATION AND ANALYSIS

The Simulation model has been designed such a way that the nodes are placed in the network in mesh topology and various protocols are been involved in the process of the simulation. The assumption made on the NS2 simulator to analyze the results of the above mentioned algorithms for group key formation, Group key generation, construction of Pythagorean Triple Tree etc.

The following assumptions are made on NS2 Simulator:

```
ns_ node-config-adhocRoutingDCMP
                -llType LL \
                -macType Mac/802_11\
                -ifqLen 50 \
                -ifqType Queue/DropTail/PriQueue \
                -antType Antenna/OmniAntenna \
                -propType Propagation/Random Way Point \
                -phyTypePhy/WirelessPhy \
                -channelType Channel/WirelessChannel \
                -topoInstance $topo
                -agentTrace ON \
                -routerTrace ON \
                -macTrace OFF
```

We have considered the different protocols such as Master-Key-Encryption-based Multiple Group Key Management (MKE-MGKM) scheme,Cipher text-Policy Attribute Based Encryption (CP-ABE), Protocols based on CertificatelessPublic Key Cryptography (CL-PKC).

The results are analysed with respect to certain parameters like

Computational Cost of group key with respect to the number of nodes in group

Group Formation time with respect to Group Size.

Percentage of Rekeying Operation performed due to a join/ leave with respect to Number of nodes in the Group.

Communicational cost of group key generation with respect to the number of nodes in the group.

4.1 Computational Cost of Group Key

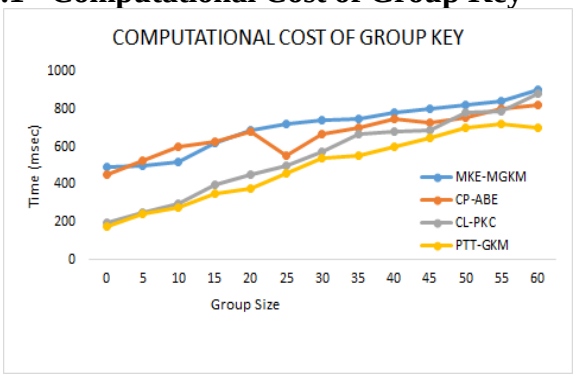

Fig 4 Computational Cost of group key

The computational cost can be evaluated as follows:

Computational Cost = Time taken (to identify node type+ Group Formation + Generating contributory key + to compute pair of keys + to compute group key) * Group Size

The Fig. 4 represents the computational cost of group key generation. The time taken gradually increases as and when the group size increases. When compared with other protocols the computational cost is less in Pythagorean Triple Tree (PTT-GKM).

4.2 Group Formation Time

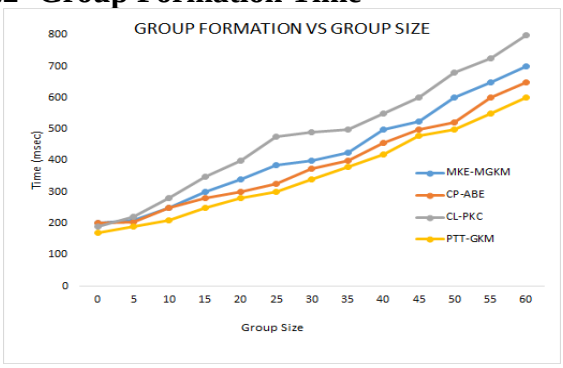

Fig 5 Group Formation Vs Group Size

The Fig 5 shows the time taken to form the group with respect to the group size. Initially the time taken to form the group is lesser than the other protocols whereas when the group size increases the time taken is same.

4.3 Communicational Cost of Group Key

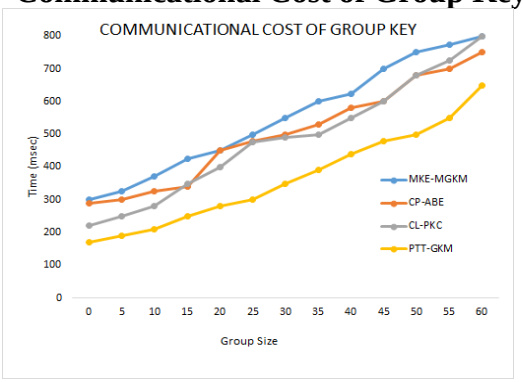

Fig 6 Communicational Cost of the Group Key

The communicational can be evaluated as follows:

Communicational Cost = Time Taken (No. of Join/Leave request messages + No. of ReplyACK + No. of key pairs (a,b) exchange) * Group Size.

The communicational cost is lesser than other protocols specified in the graph.

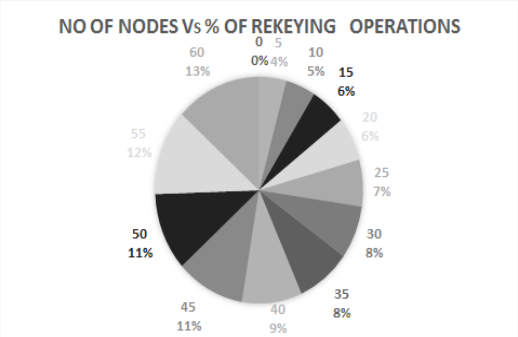

Fig 7 Number of Node VsRekeying Operation

The Fig 7 shows the percentage of Rekeying operations performed in group key generation with respect to the number of nodes in the group.

5. CONCLUSION

The secure group communication has many issues in MANETs. In this paper we have considered energy factor in Dynamic Core Multicast Routing Protocol (DCMP) to form the group. A construction of Pythagorean Triple Tree (PTT) to establish contributory key by which each node will generate a pair of keys (a, b) and it will be exchanged among the nodes to compute the group key. The group key is used to encrypt/decrypt data or information to have secure communication among the group nodes. The proposed scheme reduces the computational and communicational cost during the node joining or leaving the group. This proposed scheme is also suggested to intergroup communication that brings more scalability in the MANETs.

6. REFERENCES

[1] Anahita Mortazavi (2011) "An Efficient Distributed Group Key Management using Hierarchical Approach with Diffie-Hellman and Symmetric Algorithm: DHSA" International Symposium on Computer Networks and Distributed Systems (CNDS),PP 49 – 54, February 23-24, 2011.

[2] Beattie and Weatherby (2005), "Pythgorean Triples and Units in Integral Group Rings",Journal of Algebra and its applications, Vol 4, Issue 5, 2005.

[3] Chu-Hsing Lin and Chen-Yu Lee (2010), "Modified Autonomous Key Management Scheme with Reduced Communication /Computation Costs in MANET" International Conference on Complex, Intelligent and Software Intensive Systems PP 818 – 821, 2010.

[4] Lu Li, Ze Wang, Wenju Liu and Yun long Wang (2011) "A Certificateless Key Management Scheme in Moblie Ad Hoc Networks" 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM),PP 1-4, 2011.

[5] JikaiTeng and Chuankun (2012) "A Provable Authenticated Certificateless Group Key Agreement with Constant Rounds" JOURNAL OF COMMUNICATIONS AND NETWORKS, VOL. 14, NO. 1, PP 104-110 , FEBRUARY 2012.

[6] Min-Ho Park ,Young-Hoon Park, Han-You Jeong and Seung-Woo Seo, (2013) "Key Management for Multiple Multicast Groups in Wireless Networks" IEEE Transactions On Mobile Computing, Vol. 12, No. 9,Pp 1712-1723, September 2013

[7] NiteshSaxena and Jeong Hyun Yi (2009) "Non interactive Self-Certification for Long-Lived Mobile Ad Hoc Networks" IEEE Transactions On Information Forensics And Security, Pp 946-955 Vol. 4, No. 4, December 2009.

[8] Osamah S Badarneh and Michel Kadoch, (2009), "Multicast Routing Protocols in Mobile Ad Hoc Networks: A Comparative Survey and Taxonomy", EURASIP Journal on Wireless Communications and Networking PP 1-42, 2009.

[9] Patrick P. C. Lee, John C. S. Lui, and David K. Y. Yau (2006) "Distributed Collaborative Key Agreement and Authentication Protocols for Dynamic Peer Groups" IEEE/ACM TRANSACTIONS ON Networking, Pp - 263-277, Vol. 14, No. 2, April 2006.

[10] Wu, B., Wu, J. and Dong, Y. (2008) "An efficient group key management scheme for mobile ad hoc networks", Int. J. Security and Networks, PP 1-10, 2008.

[11] Xuanwu Zhou, Ping Wei (2008) "Key Management Scheme Based on (t, n) Threshold Cryptosystem" Proceedings of 2008 3rd International Conference on Intelligent System and Knowledge PP 1288-1293, 2008.

[12] Yi-Ruei, Tygar and Wen-Guey (2011) "Secure Group Key Management Using Uni-Directional Proxy Re-Encryption Schemes" presented as part of the main technical program at IEEE INFOCOM, PP 1952 – 1960, 2011.

[13] Zhang Chuanrong and Liu Weijiang (2010) "New ID-Based Signcryption Scheme and Its Applications in Key Update Protocols of MANET" International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery PP 254-258, 2010.

[14] Zhang, J. Zheng, and M. Ma (2008), "A Survey of Key Management in Mobile Ad Hoc Networks" HANDBOOK OF RESEARCH ON WIRELESS SECURITY, Idea Group Inc. PP 1-23, 2008.