EMICKS: - Effective Method in Cluster based Key Management in Ad Hoc Networks

K. Ramya
M.Tech
Assistant Professor / CSE,
Sree Sowdambika College of Engineering

K. Pradeepa
M.Tech
Assistant Professor / CSE,
Kalasalingam University

Abstract*— Mobile ad hoc networks are dynamically reconfigured networks in which security is a major concern. MANETS face serious security problems due to their unique characteristics such as mobility, dynamic topology and lack of central infrastructure support. Key management is crucial part of security. This issue is even bigger in MANETS. A challenge of the key distribution scheme is to use small memory size to establish secure communication among a large number of nodes. To provide good scalability in terms of the number of nodes and storage space, we utilize a combinatorial design of public-private key pairs. To reduce the overhead and for high service availability, we divide the network into clusters. Cluster head distribute the key pair. The performance results show that without clusters in a network, the metrics of overhead, time delay, key usage are more, whereas in our proposed scheme by the formation of clusters, it is greatly reduced due to its distributed behavior.

Keywords— **Key Management, Clusters, Key Update .**

I.INTRODUCTION

A mobile ad hoc network (MANET) is a collection of mobile nodes that form a temporary network without the aid of any fixed infrastructure or centralized authority. In a mobile ad hoc network, nodes communicate directly with other nodes in their transmission range and rely on other nodes to transfer messages in order to communicate with nodes outside the transmission range, in what is known as a multi-hop scenario. Security in mobile ad hoc network is considered to be more difficult than traditional networks due to the lack of infrastructure. Many security solutions rely on public key cryptography. To build a secure communication system, usually the first attempt is to employ cryptographic keys. However, cryptographic key management is challenging due to the following characteristics of wireless ad-hoc communications.

1) Unreliable communications and limited bandwidth: Due to the shared-medium nature of wireless links, flows may frequently interfere with each other. Moreover, a network may be partitioned frequently due to node mobility and poor channel condition. Therefore, the communication overhead for a certificate exchange cannot be ignored.

2) Network dynamics: Mobile nodes may leave and join the ad-hoc network Mobility increases the complexity for trust management.

3) Large scale: The number of ad-hoc wireless devices deployed at an incident scene depends on the specific nature of the incident. In general, the network size can be very large. It is necessary to have newly deployed devices and previously deployed devices trust each other without introducing too much overhead.

4) Resource constraints: The wireless devices usually have limited bandwidth, memory, and processing power. Among these constraints, communication bandwidth consumption and memory are two big concerns for key-management schemes.

Key management is an essential cryptographic primitive upon which other security primitives such as privacy, authenticity and integrity are built. However, none of the existing key management schemes are suitable for ad hoc networks. The major limitation of these schemes is that most of them rely on a trusted third party (TTP), thus not fulfilling the self-organization requirement of an ad hoc network. Special mechanisms and protocols designed specifically for ad hoc networks are necessary. Key management deals with key generation, storage, distribution, updating, and revocation and certificate service in accordance with security policies. A challenge of the key distribution scheme is to use small memory size to establish secure communication among a large number of nodes.

Due to dynamic behavior of the MANET, secret key used for communication is need to be updated whenever any node joins or leaves the network. If the network is large and also the mobility is higher, updating of the key will be more frequent. It will consume more computation power and also communication power of nodes. So in our proposal we divide the network into clusters consisting of small group of nodes. The cluster head will be elected based on the Lowest ID algorithm. If the cluster head has less energy next high capability node will be elected as cluster head. Cluster head would distribute the key pair .Compare the key distribution of without cluster network and our proposed scheme, with cluster network. The performance result show that without clusters, the computation time, time delay are more, whereas with the formation of clusters, it is greatly reduced due to its distributed Behavior.

II. RELATED WORKS

Majority of research on security of ad hoc networks emphasize the secure routing protocols, there are some proposals on key generation and distribution issues.

Zhou et al [1] proposed a technique to distribute certificate authority (CA) functionality. In this method, the networks includes n servers providing the certificates, out of which t + 1 are needed for creation of the valid certificate but t is not enough.

Seung Yi et al [2] proposed an efficient and effective distributed CA by selecting physically and computationally more secure node as MOCA (Mobile Certificate Authority) and they used threshold cryptography to distribute the CA‘s private key among these MOCA nodes.

Caner Budakoglu et al [4] proposed a modified form of distributing the certificate authority functionality. They proposed a hierarchical threshold level, so that it offers a different level of security to satisfy the needs for a wide variety of applications.

Bing Wua et al [6] propose a secure and efficient key management (SEKM) framework for mobile ad hoc networks. They build a public key infrastructure (PKI) by applying a secret sharing scheme and using an underlying multi-cast server groups. They gave detailed information on the formation and maintenance of the server groups. Each server group creates a view of the certificate authority (CA) and provides certificate update service for all nodes, including the servers themselves.

III. PROPOSED SCHEME

A challenge of the key distribution scheme is to use small memory size to establish secure communication among a large number of nodes. To provide good scalability in terms of the number of nodes and storage space, we utilize a combinatorial design of private key pairs. In our scheme, let us assume a group of people in an incident area, who want to exchange correspondence securely among each other in a pair-wise fashion. The key pool of such a group consists of a set of private key pairs, and is maintained by an offline trusted server. To support secure communication in the group, each member is loaded with all public keys of the group and assigned a distinct subset of private keys.

Consider an example of a small group with ten users. In this scheme, we need five distinct public– private key pairs to build pair-wise secure communication channels among ten users. The unique private-key set allocation for each user is then shown in Table.

In this scenario, we know that:

Each person keeps a predetermined subset of private keys, and no one else has all of the private keys in that subset;

For a public–private key pair, multiple copies of the private key

User	K_i^{priv} private-key set held by User i
1	$K_1^{priv} =\{k_1^{priv}, k_2^{priv}\}$
2	$K_1^{priv} =\{k_1^{priv}, k_3^{priv}\}$
3	$K_1^{priv} =\{k_1^{priv}, k_4^{priv}\}$
4	$K_1^{priv} =\{k_1^{priv}, k_5^{priv}\}$
5	$K_1^{priv} =\{k_2^{priv}, k_3^{priv}\}$
6	$K_1^{priv} =\{k_2^{priv}, k_4^{priv}\}$
7	$K_1^{priv} =\{k_2^{priv}, k_5^{priv}\}$
8	$K_1^{priv} =\{k_3^{priv}, k_4^{priv}\}$
9	$K_1^{priv} =\{k_3^{priv}, k_5^{priv}\}$
10	$K_1^{priv} =\{k_4^{priv}, k_5^{priv}\}$

Table I- Sample Private Key Allocation

In traditional public-management schemes, each user holds one public private key pair. Therefore, a user should store public keys and one private key to achieve self-contained key management in a network of size n. In this scheme ten-user example, a user only needs to store seven keys (five public keys and two private keys), which is smaller than 11 keys (ten public keys and one private keys) in traditional schemes. We will show that in this scheme, the total number of keys held by each user is O(log(n)), but it is O(n) under traditional key management schemes.

A. *Key Allocation Algorithm:*

```
For each node, randomly to generate the subset of keys

    If (the generated key set =)    Assigned key set.

        Move the pointer Key by Key in the generated key set to
get unassigned key set.
Else
    Assign the generated key set to node i
```

B. *Cluster formation:*

We had divided the networks into clusters and each cluster will have 1- hop nodes and cluster head. Some efficient existing algorithm can be used to group the users in to clusters and generate a cluster head for each one. The users in each cluster are in a flat network topology and the local key management policy is centralized.

The users in the group are classified into two types: cluster heads and ordinary users. The cluster head is responsible for cluster management, membership maintenance and key distribution and updating. Initially, all nodes are assigned an id, status code (for cluster head differentiation), its private key and public key. Cluster head is selected based on lowest id algorithm. To cope up with the dynamic nature of the ad hoc nodes security is enhanced by providing re-keying concept.

When we introducing the pairing concept in to clustering, each cluster could share the same key pair, so it further minimizing the key usage and also it reduce the time delay.

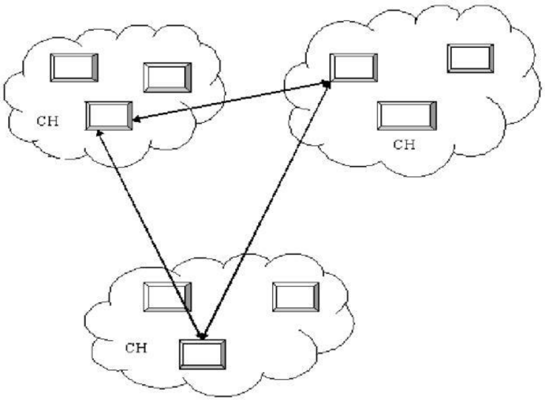


Figure1: Cluster Formation

C. Cluster head Election:-

Step 1: Assign unique ID to each node.

Step 2: Broadcast the ID of each node to the list of its neighbors.

Step 3: Elect the node as a cluster head that has the lowest ID relative to its neighbors.

D. Providing Authentication:-

Within the network, if any two nodes A and B want to communicate first it will authenticate each other. The authentication steps are as follows.

1. Node A calculate hash value using its (id, public key, private key pair) and transmit the hash value, id and public key to node B

2. Node B receive hash value and also it calculate new hash value by the information received from the node A.

3. Node B will check the received values and calculated value both are equal or not.

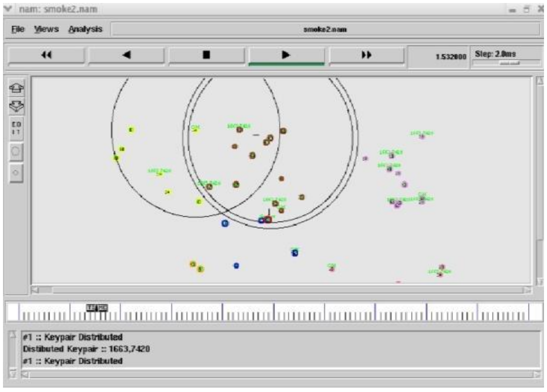
4. If the hashed values are equal, it identifies the peer node as authenticated node.

E. Algorithm for Authentication:-

```
Procedure:
BEGIN
If
    Get the peer nodes public key and ID. Calculate the hash
    value and transmit to the
peer one.
    A ->B: hash (ida, pka, private key pair) || ida ,
Pka).
Peer Node:- Calculate the same hash value. End If
If
Both hash values are same, both are authenticated nodes.
End If
END
```

IV. Implementation And Performance Comparison

The simulations are performed using Network Simulator (Ns-2), particularly in ad hoc networks. The AODV protocol is used for the simulation. Without clusters, overhead, time delay, key usage are more, whereas with the formation of clusters, it is greatly reduced due to its distributed behavior. The performance comparison is as follows:-



#1 :: Keypair Distributed
Distibuted Keypair :: 1663,7420
#1 :: Keypair Distributed

Figure2. The figure shows the result of key distribution with cluster

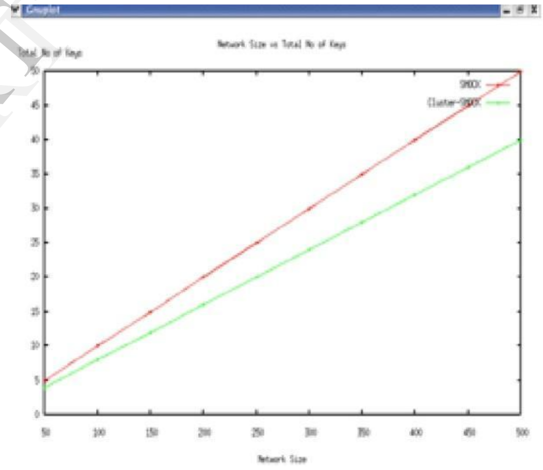


Figure3. Graphs showing the number of key usage Vs the network size

The above figure shows the tradeoff between the number of key usage and network size for a cluster network and without cluster network. The X axis represents the Network size. The Y axis represents the key usage. The number of key usage for cluster network is less while compared to the ordinary network, because by introducing the pairing concept in to clustering, each cluster could share the same key pair, so it further minimizing the key usage.

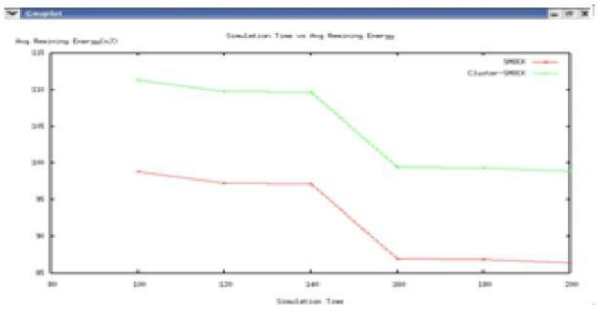


Figure4. Graphs showing the simulation Time Vs Remaining Energy.

The Energy is calculated using energy Model. Energy model is a node attribute. It represents the level of energy in a mobile node. Each node has initial value, which is the level of energy the node has at the beginning of the simulation. This is known as initial energy. It also has a given energy usage for every packet it transmit and receives. The average remaining energy is calculated by subtracting the consumed energy from the initial energy. Figure shows the tradeoff between average remaining energy and Network size. The X axis represents the Network size. The Y axis represents the average remaining energy. The average remaining energy for cluster network is less while compared to ordinary network due to its distributed behavior.

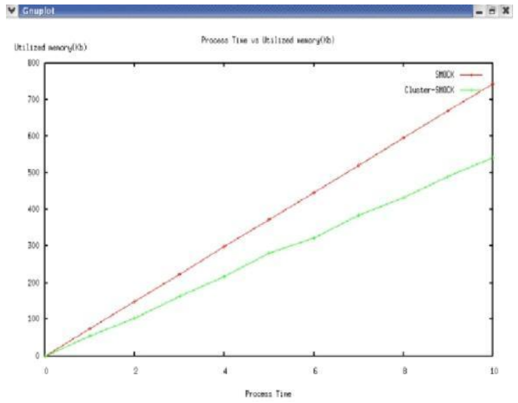


Figure5. Graphs showing the Simulation Time Vs Utilized memory.

The above figure shows the tradeoff between average simulation time and utilized memory. The X axis represents the process time. The Y axis represents the utilized memory. The utilized memory for cluster network is less while compared to ordinary network.

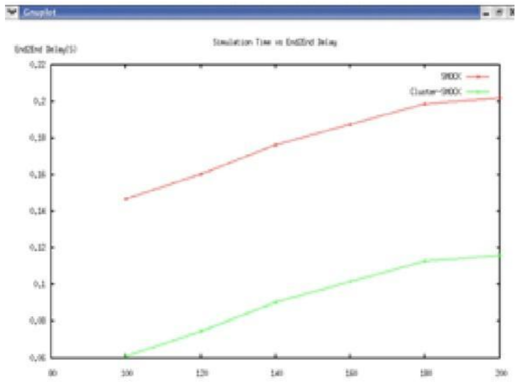


Figure 6.Graphs showing the End-End delay Vs Simulation time.

End-End delay means, time taken for a packet to be transmitted across a network. It is calculated by subtracting the transmission time from receiving time. The above figure shows the tradeoff between end-end delay of key transmission and Network Size. The X axis represents the Network size. The Y axis represents the End-End delay. The end-end delay of the packet is less in cluster network while compared to ordinary network due to its distributed behavior.

V. CONCLUSION

We depict a self-contained key-management scheme, which requires significantly less storage space than traditional schemes and almost zero communication overhead for authentication in ad-hoc network. By applying the pairing concept in to cluster network, it further minimize the key usage. And also using hash function, authentication is performed. The performance result show that without clusters, overhead, time delay, key usage are more, whereas with the formation of clusters, it is greatly reduced due to its distributed behavior.

REFERENCES

[1].Zhou, L. and Z. Haas, ―Securing Ad-Hoc Networks, IEEE Network Magazine, Vol. 13, 1999.

[2].SeungYi, Robin Craves, ―Key Management for Heterogeneous Ad Hoc Wireless Networks Proc. of the 10th IEEE International conference on Network Protocols (ICNP‘02), pages 202-203, Nov. 2002
.

[3].S.Capkun ,L.Buttyan, and J.-P. Hubaux,―Self-organized public-key management for mobile ad hoc networks IEEE Transactions on Mobile Computing, 2(1), January-March 2003.

[4].Caner Budakoglu and T.Aaron Gulliver, ―Hierarchical Key Management for Mobile Ad Hoc Networks , 0-7803-8521-7/04 IEEE, 2004.

[5].Ozkan and M.Erdem, ―Efficient Distributed Key Management for Mobile Ad Hoc Networks 0-7803-8623-W04, IEEE, 2004.

[6].Bing Wu, Jie Wu, Eduardo B.Fernandez Spyros Magliveras. ―Secure and Efficient Key Management in Mobile Ad Hoc Networks
Proceedings of the 19th IEEE International Parallel and Dis tributed Processing Symposium (IPDPS‘05) IEEE, 2005.

[7].YANG Ya-tao, ZENG Ping, FANG Yong, CHI Ya-Ping. ―A Feasible Key Management Scheme in Ad-hoc Network Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed, IEEE, 2007.

[8].Kejie Lu, Yi Qian, Mohsen Guizani, and Hsiao- Hwa Chen, ―A Framework for a Distributed Key Management Scheme in Heterogeneous Wireless Sensor Networks , IEEE Transactions on Wireless Communications ,Vol 7, Nov 2 Feb‘ 2008.

[9].Wenbo He, Ying Huang, Ravishankar Sathyam, Klara Nahrstedt and Whay C. Lee,―SMOCK: - A Scalable Method of Cryptographic KeyManagement for Mission-Critical Wireless Ad- hoc Networks IEEE Transactions on Information Forensics And Security, Vol. 4, NO 1,MARCH 2009.

[10].N.Suganthi1, Dr. V.Sumathy2 ,―An Efficient Key Management Scheme for Mobile Ad hoc Networks with Authentication ,(IJCNS) International Journal of Computer and Network Security, Vol 2, No. 5, May 2010,103.