Asian Journal of Information Technology 15 (10): 1586-1595, 2016
ISSN: 1682-3915

An Efficient Mobility Aware Group Key Management Scheme for Securing Mobile Ad-Hoc Networks

V.S Janani and M.S.K. Manikandan
Department of ECE, Thiagarajar College of Engineering, 15 Madurai, India

Abstract: In Mobile Ad-Hoc Networks (MANET), providing secure communication is extremely challenging due to the dynamic nature and the lack of centralized management. To address this issue, several researchers have proposed various key management schemes to ensure security. In this study, we describe a Mobility aware Group Key Management (MGKM) scheme for secure communication in MANETs. We present a beneficial approach in which members in the network form various groups in the same network region. An effective key tree is deployed to carry out the rekeying operation within each group, when members leave or join. Moreover, we adopt an efficient key generation scheme in order to reduce computational overhead of new keys. A revocation system that unifies direct observations with Bayesian theory and indirect observations with Dempster-Shafer (DS) evidence theory is presented to obtain a more accurate misbehaviour rate of suspicious nodes in MANETs. The proposed scheme is designed to establish a secure dynamic communication between the nodes at different regions. From our experimental results, we revealed that the proposed group key management scheme achieved higher performance and security compared with other existing key management schemes.

Key words: Group key management, MANET, mobility, rekeying, communication

INTRODUCTION

Security is a major concern in communication. Providing security for dynamic cluster based Mobile Ad-hoc Networks (MANET) with less compromising mobility and with reduced overhead is the vital task of any key management scheme. As the network is adhoc in nature, it requires an efficient system to establish a group key to allow the cluster members to communicate secretly. MANET has the characteristics such that the nodes within a communication range can freely join in and leave the cluster which makes the conventional key management schemes of ineffective (Shamir, 1984; Steiner *et al*., 2000, Khurana *et al*., 2005 and Lin *et al*., 2006). Moreover, the nodes in a MANET are usually power and energy constrained with limited network bandwidth. Consequently, a feasible scheme for MANETs must be of low computation, mobility adaptive and secured with the least number of communication rounds.

However, to resolve these inherent issues, we propose a key management scheme of Mobility aware Group Key Management (MGKM) for MANETs. The proposed scheme required no centralized authority or trustable third party. In the scheme, we adopt a tree based clustering to reduce the computation overhead, if rekeying is performed. In addition, each cluster members are capable to form a transient subgroup for secure communication, even if they originally belong to indifferent clusters. Furthermore, MGKM requires only two rounds for key generation process and one-way rekeying functionalities.

Literature review: The prime objectives of any security mechanism is to guarantee a network in terms of availability, authentication, integrity, confidentiality and non-repudiation which can be achieved with the help of key management schemes. These schemes generate keys to the nodes to encrypt/decrypt the messages and preventing the illegal usage of handling and using certified keys. That is the key management schemes facilitate highest security to networks which will manage several attacks. Most of the group key management schemes by Shamir (1984), Steiner *et al.* (2000), Khurana *et al.* (2005) and Lin *et al.* (2006) had been designed for ID-based cryptography for various group communications in a Public Key Infrastructure (PKI) system. The traditional key management systems were later found to be inactive for the dynamic MANET environment because of its varying topology and power-constrained features. Later on, researchers introduced several clusters and tree based and key management schemes by the researchers Shin and Kwon

Corresponding Author: V.S Janani, Department of ECE, Thiagarajar College of Engineering, 15 Madurai, India

(2007), Zhang *et al.* (2007), Li *et al.* (2009), Wu and Dong (2010), Li and Liu (2010) and Rahman (2008) that easily allow rekeying or key updates, key eviction and power saving. But these schemes increased the communication overhead and computation of processing key agreement. The protocols by Wang and Li *et al.* (2009) has proven methods to provide security for MANETs and provide integrity as well. The process has separate measurable modules at key generation time, secure key exchanging procedure, multilevel authentication and fast packet Coding and Decoding (CODEC) crypto analysis. These schemes were unsuccessful with a hierarchical network topology, especially when a re-keying procedure occurs.

Nevertheless, there are certain security flaws in the existing Group Key Management (GKM) mechanisms in utilizing PKI based communication system in a mobile environment. Considering the special features of MANETs such as mobility that members leave or join the cluster, we need to assess the group key management protocols for MANETs. Moreover, owing to the presence of topology, providing a promising secured group key management in MANET is difficult to achieve. We propose an efficient Mobility Aware Group Key Management Scheme where a key tree is deployed in a consistent rekeying operation whenever members leave or join. In addition, we adopt an efficient key generation scheme to reduce computational overhead of new keys, after each mobility event.

MATERIALS AND METHODS

This section describes the key management functionalities of MGKM.

Network model: In the proposed scheme, we assume the mobile nodes are grouped into clusters based on location information of each node, obtained from the Location Based Multicast (LBM) protocol by Ko and Vaidya (1999) employed in the network. The nodes in the same geographic location are considered as neighbours, to deploy group keys. The size of the clusters may change dynamically with nodes joining, leaving or failing over time.

Key initialization: Initially a virtual binary tree is constructed among the members in each cluster. To reduce the complexities and cost of computation, we consider the fact that the set of tree-branch of a particular node is the subset of tree-branch of its parent node. Each member in the cluster has an asymmetric key pair, besides the node's subset shares the internal keys. The public key

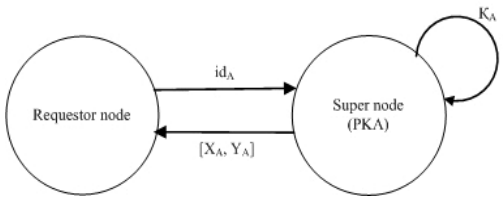

Fig. 1: Key generation in MGKM

of each member is represented by the leaf keys. The leftmost nodes are prioritized as supernodes or clusterhead to adapt the mobility.

Consider a MANET with N super nodes, referred as Key Generating Authority (in MGKM which acts in the role of certificate authority. Let $\{n_1, n_2, ..., n_k\}$ are set of nodes located in a cluster of the same location by which key tree is built. The key tree is updated orderly with the addition of new nodes. The leftmost node in the key tree is considered as supernode which is responsible for key generation in each cluster. The supernode runs a Bilinear Diffie-Hellman (BDH) generator for pairing two true groups, namely A and B of order level γ. Let g_A and be the generator of group A and B respectively, y and h be the hash value used for mapping keys.

Key generation: The MGKM scheme allows two nodes to establish a group session key for encryption. A two party authenticated key agreement scheme using bilinear pairing is used to generate a pairwise key as shown in Fig. 1. The supernode (also called) chooses a random number n as its private key, where $n \epsilon \{0, 1\}$. The public key is calculated as $K_{pub} = ng_A$. When a member in the tree of the same cluster sends its identity to the supernode, it then computes the hashing value $_A = h(id)_A$ where id_A is the unique id of the requestor nodes. The supernode then computes the pairwise keys: $x_A = (A+n)g_A$ as public key and $y_A = (A+n)^{-1} g_A$ as private key and sends to the requestor node. The supernode preloads the pairwise keys of every member node securely before each key exchange.

Key establishment: In MGKM, we present a secured key establishment procedure to compute a group session key as in algorithm 1 and 2.

Algorithm 1: pairwise session key establishment:

n_i: sender node, subset of supernode S_A of group A
n_i: receiver node, subset of supernode S_B of group B
Where n_i, $n_j \in A$

S_A chooses a short-term key (where = {0, 1} randomly

S_A computes a short-term pairwise session key $P_{i,j}$ with the hash function, public key and the short term key as:

$$P_{i,j} = \left({}_A g_A + K^i_{pub} \right) \quad (1)$$

S_A establishes a Lagrange interpolation polynomial by Weisstein, 2001, $L_{i,j}$ with session keys of S_A and S_B

S_A broadcasts $L_{i,j}$ to all the supernodes in the cluster based network

S_B uses its short-term pairwise session key S_A and $j_{j,i}$ and recover from $L_{i,j}$ to establish a pairwise session key between cluster A and B

Algorithm 2: Group session key establishment:

Each supernode that hear the $L_{i,j}$ uses their short-term pairwise session keys to recover the short-term key from $L_{i,j}$

The supernodes compute a group session key after recovering as:

$$GSK = (q_1 + q_1 + ... + q_k) g_A U_p \quad (2)$$

Where U_p: number of key update phases predefined by the KGA

Proposed group key revocation: The revocation of compromised keys is carried out different phases, namely misbehaviour reporting, verification and revocation.

Misbehaviour reporting: To keep track of the behaviour of each node in the tree, the supernode classifies the behaviour into acceptable, suspicious and mischievous and includes nodes into three different lists namely white, grey and black lists respectively. The nodes in the whitelist are considered as trustable nodes. The suspicious nodes can be either a selfish node or a compromised one whereas the mischievous nodes are compromised nodes. A cluster based monitoring system is employed in each group by which each node can watch its 1-hop neighbour's behaviour. Let $_A = \{n_1, n_2, ..., n_i\}$ be the set of 1-hop neighbours of node k in group A. Each node in the clustered network maintains a behaviour matrix denoted by to record the behaviour of its 1-hop neighbours. The observations are refreshed at regular intervals for adapting mobility.

The behaviour matrix includes the accusation factor (α^k) that the node k creates on its neighbour from the behaviour evidences, the binary variable (b^v) of nodes in ^{A} and their public keys. The accusation factor gains some property: It lowers the trust of a node when it is misbehaved. The trust of the misbehaved node cannot be recovered quickly even though it forwards large true packets. The accusation factor $\alpha^k = \{0, 1\}$ is set to 1 if a node is suspicious and set to 0 if else. The behaviour matrix is given as:

$$_k = \begin{bmatrix} id_1 & k^1_{pub} & b^1_v & \alpha^k_1 \\ id_2 & k^2_{pub} & b^2_v & \alpha^k_2 \\ & \vdots & & \\ & \vdots & & \\ id_i & k^i_{pub} & b^i_v & \alpha^k_i \end{bmatrix} \quad (3)$$

When α^k of any node in $_A$ is set to 1, the observer k sends a misbehaviour alert message (k) to the supernode secretly. The alert message includes the suspicious node's accusation factors with its public key. For example, consider n_i is the suspicious node, suspected by k, then $_k$ is given as:

$$_k = \alpha^k_i (id_i) k^i_{pub} \quad (4)$$

Revocation system with bayesian and evidence theorem: When the supernode hears the misbehaviour alert message from any node, it verifies whether the message is attained from an acceptable node (ie., a whitelisted node). The supernode puts the suspected node in grey list and requests the 1-hop neighbours of the suspected node (say x) to share their independent observations about x. We consider the observations as evidences that is in the form of number of observed misbehaviours to calculate the evidence accusation factor ($\alpha^x(e)$). The supernode also computes the misbehaviour rate in terms of accusation factor ($\alpha^x(d)$) by directly observing the node x. The revocation systems, usually combines the direct observations and the evidences obtained from the one-hop neighbours to decide the trustworthiness of x.

Many of the existing revocation systems let-down when the observing node itself is untrustworthy which contributes no true evidences. Such systems might be impracticable especially to inform which observer node is untrustworthy. Hence, we use Dempster-Shafer (DS) evidence theory developed by Dempster and revised later by Shafer where the uncertainty is represented in the form of belief functions. The core idea of the DS theory is that an observer acquires a certain degree of belief on a proposition based on the subjective probability of a related proposition or hypothesis. DS theory aims to provide a convenient mathematical model to combine disparate information obtained from different sources.

Misbehaviour verification with Bayesian theory: We assume that the supernode can watch the key forwarded by the suspicious node and compare them with the original packets to identify the misbehaviour nature of the node x. Therefore the supernode directly calculates the accusation factor of its member nodes by Bayesian inference where the unknown probabilities are inferred using observations. The measure of belief about a proposition or hypothesis is stated with well known Baye's theorem:

$$P(p|e) = \frac{P(e|p)P(p)}{P(e)} \quad (5)$$

Where:

[p/e] = Measure of belief about the proposition (p) with respect to the evidence (e)

P[p] = belief about in the absence of e

The Baye's theorem can also be expressed in terms of probability distribution as:

$$P(\phi|data) = \frac{P(data|\phi)P(\phi)}{P(data)} \quad (6)$$

Where:

[ϕ|date] = Posterior distribution for the parameter ϕ
P[date|ϕ] = Sampling density function
P[ϕ] = Prior distribution
P[data] = Marginal probability function of data

From (2), we can modify the misbehaviour verification as:

$$P(\phi,i|j) = \frac{f(j|\phi,i)P(\phi,i)}{\int_0^1 f(j|\phi,i)P(\phi,i)d\phi} \quad (7)$$

Where:

ϕ = Degree of belief and $0 \le \phi \le 1$
j = Rate of correctly forwarded keys by a node
I = Rate of keys received by the node
f(j\ϕ, i) = Probability function that follows a binomial distribution given by

$$f(j|\phi,i) = \binom{i}{j}|\phi^j(1-\phi)^{i-j} \quad (8)$$

To describe the initial knowledge concerning probabilities of success, we use beta distribution to the Bayesian approach and hence the prior distribution P(ϕ, i) can be stated as:

$$(\phi;\alpha,\beta) = \frac{\phi^{\alpha-1}(1-\phi)^{\beta-1}}{\int_0^1 f(j|\phi,i)P(\phi,i)d\phi} \quad (9)$$

Where α, β>0 is the power function of i and j. The mean and variance of the beta distribution function is given as:

$$E(\phi|\alpha,\beta) = \frac{\alpha}{\alpha+\beta} \quad (10)$$

$$V(\phi|\alpha,\beta) = \frac{\alpha\beta}{\alpha+\beta+1} * \frac{1}{(\alpha+\beta)^2} \quad (11)$$

In our scheme the accusation factor reflects the behaviour fading thereby giving more weights on the misbehaving rate in Bayesian network. The accusation factor for misbehaviour verification is given as:

$$E(\phi|\alpha,\beta) = \frac{\alpha}{\alpha+\alpha^x\beta} \quad (12)$$

On considering the transaction history in the Bayesian framework for misbehaviour calculation, the expectation of beta distribution can be written as:

$$E(\phi|\alpha,\beta) = \frac{\alpha_t}{\alpha_t+\alpha_t^x\beta_t} \quad (13)$$

Where:

α_t = $\alpha_{t-1}+i_{t-1}$
β_t = $\beta_{t-1}+j_{t-1}$ and initially no observations are made and so $\alpha_0, \beta_0 = 0$

Based on the above deduction, we compute the direct accusation factor of the supernode on node x can be written as:

$$\alpha^x(d) = E(\phi|\alpha,\beta) \quad (14)$$

Misbehaviour verification with evidence theory: This section describes the misbehaviour verification based on the indirect observations from the 1-hop neighbours of the suspicious node x.

As shown in Fig. 2, the supernode requests the 1-hop neighbours of x to verify the misbehaviour rate from their independent observations about x. The observations (also called evidences) obtained from the 1-hop neighbours help in judging the trustworthiness of x. To perform this, the DS theory is used with uncertainty or ignorance. This theory is based on key element namely belief function which depends on the subjective propabilities that are combined to form indirect evidences.

In DS evidence system, the probabilities which are mutually exclusive and exhaustive are considered as a set and considered as frame of discernment, denoted by Ω as introsuced by Wu *et al*. (2009). A power set represented by 2^Ω includes all basic probabilities of the proposition called focal values A_k which is a function of m and satisfies the following conditions:

- Basic probability value of null set is zero, ie., $m(\phi) = 0$
- Sum of elements in 2^Ω is 1, ie., $\sum_{A_k \subseteq \Omega} m(A_k) = 1$

The belief function can therefore defined as:

$$B(x) = \sum_{A_k \subseteq x} m(A_k) \quad (15)$$

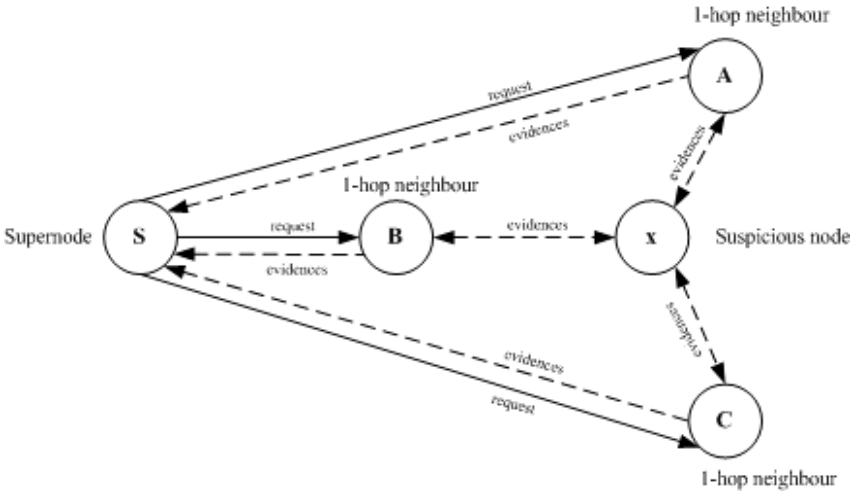

Fig. 2: Indirect misbehaviour verification

In the proposed scheme we designed two behaviour states to the nodes ie., {accept, revoke} demonstrated with DS theory. The frame of discernment consists of two possibilities concerning the behaviour level for any random node as = Ω{ accept, revoke}that represents a) an accept or allowable behaviour state b) a revoke or mischevious state. On considering Fig. 2, the 1-hop neighbours node A, B and C of x shares their evidences as a subset of Ω. The power set 2^{Ω} includes three possible forms of hypothesis ie., hypothesis A = {accept}, hypothesis R = {revoke} and hypothesis H = Ω that represents node x is either in acceptable or revoked state. The 1-hop neighbour provides evidences based on their direct observations by sharing its belief over Ω. For example, if node A believes x behaves trustworthy then $m_A(A)$ is α^x and therefore $m_A(R)$ is 0. From the key management scheme by Wu *et al*. (2009), the evidence from node A can be stated as:

$$m_A(A)=\alpha^x(A), m_A(R)=0, \quad m_A(H)=1-\alpha^x(A) \tag{16}$$

Likewise, if node B believes x as misbehaved, its evidence favours revoke function as follows:

$$m_B(A)=0, m_B(R)=\alpha^x(B), \quad m_B(H)=1-\alpha^x(B) \tag{17}$$

DS theory of combining evidences: The DS theory combines all the 1-hop neighbours evidences based on the condition that the evidences are independent as presented by Chen and Venkataramanan, 2005. Suppose $B_1(x)$ and $B_2(x)$ are two independent observer's belief functions over same suspicious node, then the orthogonal sum of these functions is given as:

$$B(x)=B_1(x)+B_2(x)=\frac{\sum_{j,k,A_j\cap A_k=x} m_1(A_j)*m_2(A_k)}{\sum_{j,k,A_j\cap A_k\neq\phi} m_1(A_j)*m_2(A_k)} \tag{18}$$

Where A_j, $A_k \subseteq \Omega$. With reference to Fig. 2, the belief of node A and B is calculated as given by Chen and Venkataramanan, 2005:

$$\begin{aligned} m_A(A)\oplus m_B(A)&=\frac{1}{1}\left[\begin{matrix} m_A(A)m_B(A)+ \\ m_A(A)m_B(H)+m_A(H)m_B(A)\end{matrix}\right] \\ m_A(R)\oplus m_B(R)&=\frac{1}{1}\left[\begin{matrix} m_A(R)m_B(R)+ \\ m_A(R)m_B(H)+ \\ m_A(H)m_B(R)\end{matrix}\right] \\ m_A(H)\oplus m_B(H)&=\frac{1}{1}\left[m_A(H)m_B(H)\right] \end{aligned} \tag{19}$$

$$\begin{aligned} 1=m_A(A)m_B(A)&=m_A(A)m_B(H)+ \\ m_A(H)m_B(H)&+m_A(H)m_B(A)+m_A(H) \\ m_B(R)&+m_A(R)m_B(R)+m_A(R)m_B(H) \end{aligned} \tag{20}$$

In MGKM, we assume the acceptance rate of probability of node A and B is 0.8 and 0.7, respectively and therefore B(A) = 0.94, B(R) = 0, B(H) = 0.6. Thus, the acceptable behaviour value from the indirect observation with DS theory is 0.9. In general, the evidence accusation factor obtained from the indirect observations can be computed as:

$$\alpha^x(e)=B(x) \tag{21}$$

Revocation in MGKM: The revocation process in MGKM is given in algorithm 3.

Algorithm 3: Revocation in MGKM:

When the supernode hears the misbehaviour alert message from any node, it verifies whether the message is attained from an acceptable node

Supernode (S) puts the suspected node (say, $x \epsilon_A$) in grey list

S calculates the accusation factor by directly observing the node x and requests the 1-hop neighbours of x to send their independent observations about x

S considers the observations of 1-hop neighbours as evidences on the behaviour of x

S computes the total accusation factor (À) and checks whether À is greater than the security parameter γ where γ is the misbehaviour threshold set by the supernode above which the supernode assumes an unsecured communication

$$A_x = \lambda \alpha^x(d) + (1-\lambda)\alpha^x(e) > \gamma \quad (22)$$

Where λ is the weight assigned to direct observation and $0 \le \lambda \le 1$

If $A_x > \gamma$, S sets the node x in the black list and revoke its pairwise key

S floods a revocation message REV $\langle id_x, K_{pub}^x \rangle$ throughout the network for intimating other cluster nodes

Mobility aware group key management: The MGKM scheme is designed to support key functionalities even in the presence of node's mobility.

Member joining and eviction: In MANET, new nodes join and evict dynamically and so the tree-based cluster changes frequently. Even the supernode joining or eviction occurs in the ad-hoc network. A node eviction can happen due to several reasons like communication failure, unavailability or revocation. The member joining and eviction can change the key structuring and key functionalities, especially key generation and distribution which increases the computational complexities and cost. Hence rekeying should be computed in such a way that the computational cost and complexities should be degraded.

Rekeying in MGKM: We propose a one-way key generation procedure in MGKM whenever a node joins or leaves the cluster. This one-way key generation is employed in such a way that the supernode need not involve to initialize the key update procedure each time. This reduces the communication rounds and overheads in key distribution, making the network flexible for mobility. Initially, when a group tree is constructed, the supernode computes the hash value for each of its members. When a new node joins the cluster, it is inserted into the key tree first. The primary keys are refreshed with secondary keys along the route from the new node to the supernode in the next step. The new key is generated from the primary key as:

$$x = f\left(x \oplus_A^S\right) \quad (23)$$

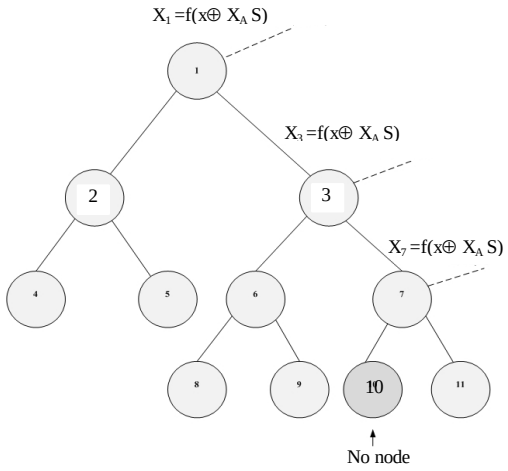

Fig. 3: Rekeying for member joining

Where:

x = Secondary or new key

x = Primary or old key

s = Non-zero cryptographic salt value

The cryptographic salt value as shown by Ko and Vaidya (1999) which is generated at random to lower the probability of hash function will be utilized from any pre-defined key table, providing forward and backward secrecy to the cluster based MANET. The rekeying operation when a node joins a cluster is shown in Fig. 3.

Likewise when a node leaves the cluster, it sends a resign message with its pairwise session key to supernode. The supernode verifies the resign message, whether it is true or not. The supernode then broadcasts a delete message to all its members along with the new hash variable A. A new key is computed along the path from the resigned node to the supernode with new hash and salt function as:

$$x = f\left(x \oplus_A^S\right) \quad (24)$$

The rekeying operation when a node evict from a cluster is shown in Fig. 4.

Security model: Forward secrecy: when a node leaves a secured group, it cannot access the new key which is referred as forward secrecy. This is achieved by the rekeying operation performed by MGKM. The departed node cannot access the new key as well as the new one way hash function with the salt value.

Backward secrecy: the backward secrecy prevents the new node from exploiting the primary key used by the cluster tree, previously. This is because the cluster performs the rekeying operation that deletes the old keys which cannot be utilized and recognized by other nodes.

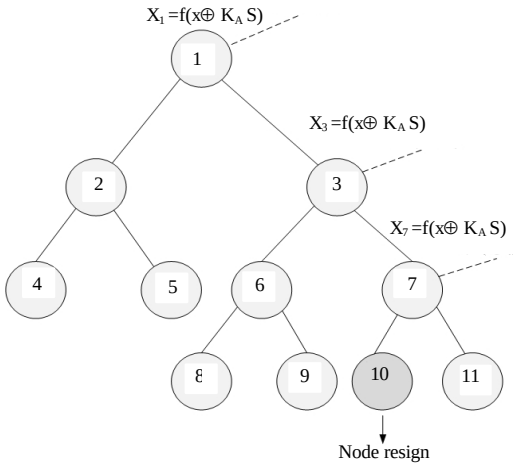

Fig. 4: Rekeying for member eviction

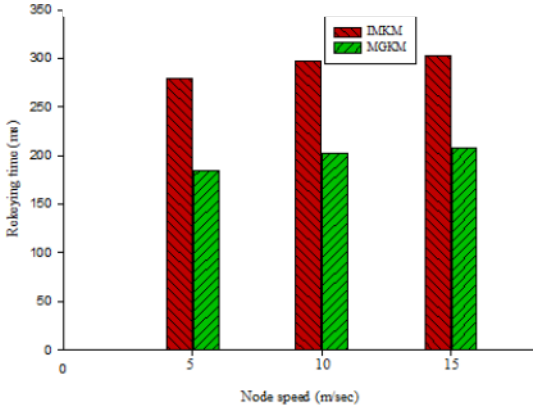

Fig. 5: Rekeying completion time

Key security: during each execution of the MGKM, a unique group session key is generated with short term keys chosen randomly. Moreover, the compromised group session key used previously doesn't affect the future key processes.

Key authentication: the supernode computes the pairwise group key for the members using short term keys and private keys. This prevents other external nodes from obtaining the group key which can only be accessed with the member's private keys.

Confidentiality: in MGKM, all the messages in the cluster are encrypted by the group session key computed by the key generation process. Without knowing the group key, the decryption of messages by the adversary cannot be possible in a secured group communication. The confidentiality depends on the group key secrecy guaranteed by forward and backward secrecy which is established by the MGKM.

RESULTS AND DISCUSSION

The MANET simulation setup is performed in Qual Net 4.5 environment with IDE: visual studio 2013, programming language: VC ++ and SDK: NSC_XE-NETSIMCAP (Network Simulation and Capture). Comparison of two different schemes is run in a simulation environment of 40 nodes that follow a Random Walk Mobility (RWM) model presented by Bai and Helmy in which each node changes its mobility rate at different time intervals. In this simulation, the bandwidth/channel capacity of all mobile nodes is assigned as 4 MBPS. The simulation environment is configured to evaluate the performances of both the key management schemes with mobility robustness, in terms of overhead (in respect of the average number of messages sent and number of exponents), key update time and other performance parameters (communication round, number of pairing, memory, strength of security and power consumption) as shown in the figures below.

Comparison in rekeying: We compare our MGKM with IMKM by Li and Liu (2010) with respect to rekeying (or key updates) and key establishment. For IMKM, the cluster leader broadcasts a key update message to other key generators after which key eviction process gets initialized. Whereas for MGKM, the rekeying operation or key update is carried out whenever a node joins or evict the cluster, using a one-way, light weighted key generation process. The nodes in the path from the newly added or evicted node to the supernode directly involve in the key update process, wherein the supernode need not broadcast update message and initialize the operation. This reduces the rekeying completion time in MGKM as shown in Fig. 5. The average completion time for rekeying process of 40 nodes at different node speed is simulated in Fig. 5.

Overhead comparison: We also count the key management overhead in terms of number of messages and number of exponents which includes all the key requests and replies in the MGKM and IMKM schemes as shown in Fig. 6a and b. From Fig. 6a it is observed that overhead is the same at all node mobility, making both the schemes robust to dynamic mobility. However, IMKM requires larger overhead compared to MGKM scheme. The communication overhead in terms of number of exponents for different number of nodes is shown in Fig. 6b wherein MGKM possess less overhead compared to IMKM for larger number of nodes.

Revocation management system: In our simulations, we assume two types of nodes in the network: normal nodes

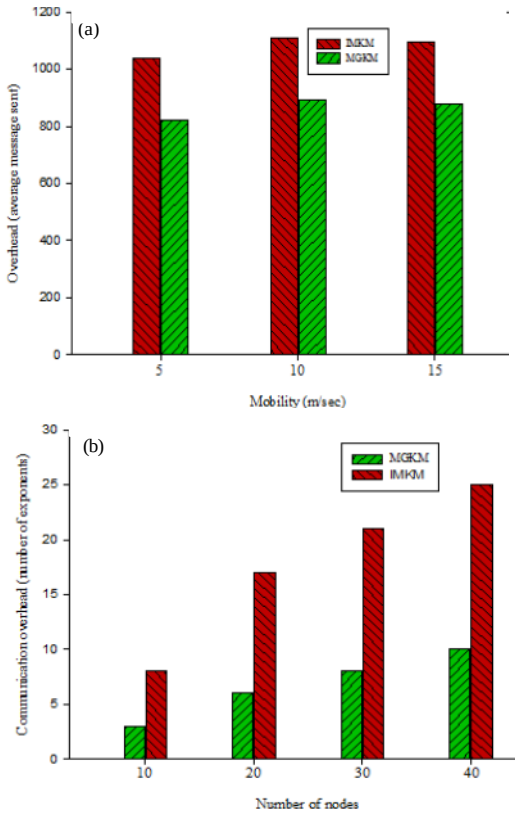

Fig. 6:Overhead comparison; a) average message sent; b) communication overhead

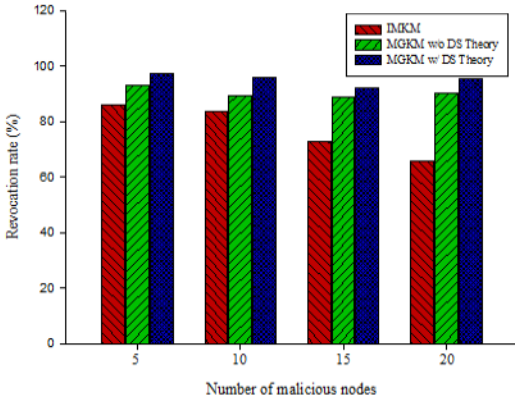

Fig. 7 : Revocation rate

which follow are trustable and misbehaved nodes which modify or drop keys mischievously. We assume that the number of misbehaved nodes is less than 50% of the total number of nodes in the network. In this adversary state, the proposed scheme is evaluated and compared with the IMKM protocol. We have simulated the revocation system with different numbers of nodes. The RWM model developed by Bai and Helmy, 2004 is adopted in a

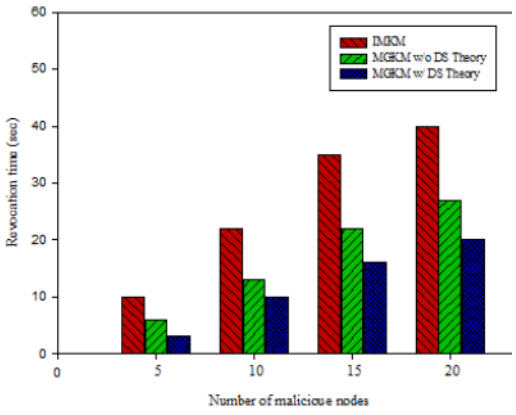

Fig. 8: Revocation time

40-node MANET with a maximum velocity of nodes is set to 0-10 m^{-1}s and 30 sec pause time. We consider three performance metrics in the simulations, to analyse the effectiveness of the proposed revocation system: revocation rate which represents the rate of attacker nodes revoked before launching the attacks; revocation time which is the time taken for revoking the keys of misbehaved nodes before launching an attack; verification overhead which includes the request for observations, evidence and accusation factor computation in MGKM and IMKM schemes.

In Fig. 7, we compare the rate of revocation in MGKM with and without DS Theory and original IMKM, where the malicious nodes vary from 5-20. As shown in Fig. 7, the proposed scheme has a much higher revocation rate than the existing scheme because the proposed revocation system can detect the misbehaviour of nodes. The result also reveals that MGKM with DS Theory has the highest revocation rate among the three schemes. It is also noted that the revocation rate of the proposed MGKM scheme with and without DS Theory varies slightly for higher number of nodes. Whereas the IMKM scheme decreases the rate of revocation moderately with the raise in the number of nodes. This is because the collision of revocation messages sent becomes more frequent with the increase in the number of nodes in the MANET. Although the revocation rate varies in three schemes, the proposed MGKM scheme is evidently better than the existing IMKM.

In Fig. 8 the revocation time for three schemes are compared. Revocation time is a crucial factor for estimating the performance of revocation strategy. For analysis sake, we kept the number of malicious nodes from 5-20. The first observation in the revocation time is small for MGKM when compared with IMKM, due to close monitoring of the nodes in MANET. A restrained increase in the revocation time can be noted from the

Table 1: Comparison of key management schemes

Schemes	Communication rounds	Number of pairing computations	Memory	Power consumption(mW)	Security level(%)
Multiparty GKA by Lin	2	2^x	x-1	1207	79
SEGK by Wu	2	2^x	2x-2	1114	81
MGKM	1	x	(2x/g)-2	860	87

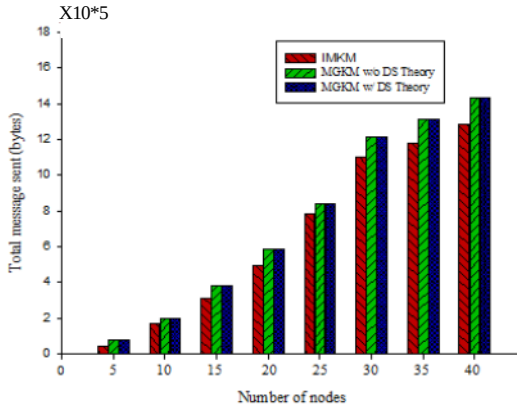

Fig. 9: Total message sent

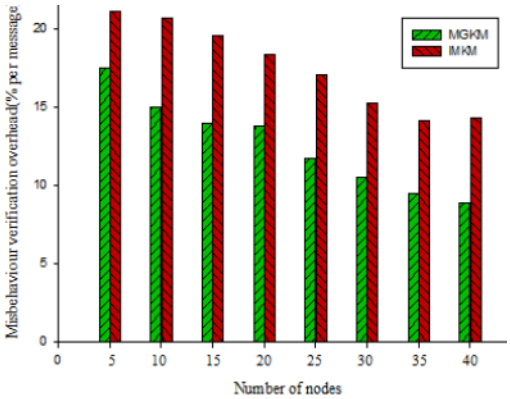

Fig. 10: Misbehaviour verification overhead

MGKM scheme whereas the time increases highly in IMKM with the increase in the number of malicious nodes. This is mainly because as the number of malicious nodes raises, the evidences also increases to identify the misbehaviour hence the revocation time increases.

As key management computation, sending and receiving of messages are also an important issue in MANETs which consumes large amounts of energy. Thus, we the amount of overhead of messages imported when the misbehaviour verification is done in both INKM and MGKM schemes. Fig. 9 shows the rate of message overhead that is incorporated compared with the original IMKM. Because the accusation factors are combined using DS Theory, no additional messages are needed to be sent. The overhead in MGKM is not very high. As in Fig. 10 with the increase in number of nodes, the percentage of overhead in misbehaviour verification drops dramatically. This is because when the number of nodes grows, the total verification message becomes large which includes evidence and accusation factor computation.

Performance Comparison: We compare MGKM scheme with two other schemes- Multiparty Key Agreement by Lin *et al.* (2006), key agreement scheme and Simple and Efficient Group Key management scheme (SEGK) by Wu *et al.* (2009) in terms of cost of computation and communication as shown in Table 1. Let be the total number of nodes and be the number of MGKM groups. We use the following factors:

- Communication round: total number of communication rounds that occur when a node joins and evict the cluster
- Pairings: total number of pairing
- Memory: total memory to be stored for each key process
- Power consumption: average power consumed for key management procedures
- Security level: the level of security achieved by each scheme in providing the security models described in section V

From the observations, it can be stated that MGKM has better performance than the other protocols. Because our scheme shows an absolute advantage in the number of communication round, pairing computation and average memory compared with key management schemes developed by Lin *et al.* (2006) and Wu *et al.* (2009) which can reduce the cost computation and communication. Furthermore, MGKM achieves a security level of 87% with less power consumption, comparing with other schemes.

CONCLUSION

Secured key management in MANET is much more difficult than other classical networks owing to the number of nodes and the lack of infrastructure. In this, we have addressed a secure and efficient Mobility Aware Group Key Management (MGKM) scheme for cluster based mobile ad-hoc networks. In order to adapt the high mobility and varying link qualities of MANET whenever

members leave or join, the rekeying operation is efficiently carried out. In contrast to the existing techniques, we have proposed MGKM to efficiently rebuild (log (n)) keys whenever a node leaves or joins the cluster. To reduce the overhead of communication in key distribution and bandwidth usage, we present a one-way key generation technique that satisfies all security concerns. In this, we have presented a unified revocation management scheme that strengthens the security of MANETs. We evaluated the accusation rate of suspected nodes using the uncertainty reasoning theories such as Bayesian inference and DS Theory, where misbehaviours can be detected through direct and indirect observation. From the analysis, we believe that the proposed MGKM scheme improves on the security and performance (in terms of mobility robustness, communication overhead, key update time, computation and communication cost) of previously proposed key management schemes.

RECOMMENDATIONS

Some future directions in key management are worth investigating. First, we will consider effective ways to join keys with verifiable secret sharing inorder to reduce the exposure of shares. Second, it is interesting to analyse distributed algorithms to revoke malicious nodes without using the supernode concept.

ACKNOWLEDGEMENT

This research is supported by All India Council for Technical Education (AICTE), Government of India.

REFERENCES

Chen, T.M. and V. Venkataramanan, 2005. Dempster-shafer theory for intrusion detection in ad hoc networks. Internet Comput. IEEE., 9: 35-41.

Khurana, H., R. Bonilla, A. Slagell, R. Afandi and H.S. Hahm *et al.*, 2005. Scalable Group Key Management with Partially Trusted Controllers. In: Networking-ICN 2005. Pascal, L. and P. Dini (Eds.). Springer Berlin Heidelberg, Berlin, Germany, pp: 662-672.

Ko, Y.B. and N.F. Vaidya, 1999. Geocasting in mobile ad hoc networks: Location-based multicast algorithms. Proceedings of the 2nd Workshop on Mobile Computer Systems and Applications, February 25-26, 1999, New Orleans, LA., pp: 101-110.

Li, L.C. and R.S. Liu, 2010. Securing cluster-based ad hoc networks with distributed authorities. IEEE Trans. Wireless Commun., 9: 3072-3081.

Li, M., Z. Feng, N. Zang, R.L. Graham and F.F. Yao, 2009. Approximately optimal trees for group key management with batch updates. Theor. Comput. Sci., 410: 1013-1021.

Lin, C.H., H.H. Lin and J.C. Chang, 2006. Multiparty key agreement for secure teleconferencing. Proceedings of the 2006 IEEE International Conference on Systems, Man and Cybernetics, SMC'06, October 8-11, 2006, IEEE, Taipei, Taiwan, ISBN: 1-4244-0099-6, pp: 3702-3707.

Rahman, R.H. and L. Rahman, 2008. A new group key management protocol for wireless ad-hoc networks. Int. J. Comput. Inform. Sci. Eng., 2: 74-79.

Shamir, A., 1985. Identity-Based Cryptosystems and Signature Schemes. In: Advances in Cryptology, Blakley, G. and D. Chaum (Eds.). Springer, Berlin, Heidelberg, ISBN: 978-3-540-15658-1, pp: 47-53.

Shin, S. and T. Kwon, 2007. Efficient and secure key agreement for merging clusters in ad-hoc networking environments. IEICE Trans. Commun., E90-B: 1575-1583.

Steiner, M., G. Tsudik and M. Waidner, 2000. Key agreement in dynamic peer groups. IEEE Trans. Parallel Distrib. Syst., 11: 769-780.

Wu, B. and Y. Dong, 2010. A simple group key management approach for mobile ad hoc networks. Proceedings of the 5th 2010 IEEE International Conference on Networking, Architecture and Storage (NAS), July 15-17, 2010, IEEE, Macau, China, ISBN: 978-1-4244-8133-0, pp: 73-78.

Wu, B., J. Wu and Y. Dong, 2009. An efficient group key management scheme for mobile ad hoc networks. Intl. J. Secur. Networks, 4: 125-134.

Zhang, J., L. Sun, Y. Tang and S. Yang, 2007. D-VKT: A scalable distributed key agreement scheme for dynamic collaborative groups. IEICE Trans. Commun., E90-B: 750-760.