Secure and Highly Efficient Three Level Key Management Scheme for MANET

Wan An Xiong, Yao Huan Gong
School of Electronic Engineering
University of Electronic Science and Technology of China (UESTC)
Cheng Du, si chuan, China
xwa985@163.com

Abstract: - MANET(Moving Ad hoc Network) is a convenient infrastructure-less communication network which is commonly susceptible to various attacks. Many key management schemes for MANETs are presented to solve various security problems. Identity (ID)-based cryptography with threshold secret sharing ,ECC and Bilinear Pairing computation is a popular approach for the key management design. In this article, we adopt these approaches to construct tree structure and cluster structure ad hoc network which has three level security communication framework. After constructing the security structure, we evaluate the security performance and efficiency of the scheme in detail.

Key-Words: - Three Level Key Management, Elliptic Curve Cryptography, Bilinear Pairing Computation, (n,t) Threshold Key Distribution, ID-based key management

1 Introduction

A mobile ad-hoc network (MANET) is formed on-the-fly and it is also a convenient infrastructure-less communication network. So we can construct MANET on demand without support from central servers. MANETs are especially suitable for communications in critical situations such as battlefield, emergency and rescue missions. In the mean time, MANETs are highly vulnerable to various security threats , because they have the following inherent characteristics: open medium, absence of fixed central structure, dynamically changing topology, constrained capability, etc. So the research on key management schemes becomes the focus of attention, but so far, most of the existing schemes are either incomplete or not appropriate for small resource constrained devices.

MANETs are peer network and each node has equal effect. So, on the one hand, we should construct distributed KDC(key distributing centre), on the other hand, when the size of MANETs increase, node join and node leave will result in all MANETs nodes' key update. This will bring some problems such as traffics and computations increase. In order to solve this problem, we must classify all nodes by different security levels. So, we can get the following network topology structure: all nodes are classified as multi-cluster and the cluster head is selected. The cluster head is responsible for special tasks such as key management and trust management. The detailed function of cluster head can be confirmed by designer. Literature [1] described three categories methods cluster head often adopt in node secure communication. They include: 1) All nodes communicate with symmetric cryptography, nodes apply unified key to realize authentication. Cluster head is only used as key distribution. 2) Cluster head take on the task of distributing CA(certificate authentication), it can produce and distribute certificate by getting together with other nodes. Cluster head node can also provide all certificate functions for its internal nodes. 3) As a node of distributing CA, cluster head can issue certificate for all nodes of the MANETs. These certificates can be authenticated between different clusters. Nodes within one cluster apply symmetric key to realize communications. Nodes which belong to different clusters should be cross authenticated via different cluster heads. This method can make the ad hoc network adapt to the dynamic topology efficiently, and key management and authentication scheme be neatly designed.

Based on multitudinous ECC(Elliptic Curve Cryptography) and IBE(Identity Based Encryption) schemes, we apply bilinear pair computation to realize secure key management and communication.Then with shamir's (n,t) threshold key management scheme, we build three level security ad hoc network. This scheme can be neatly

applied to the dynamic topology and different sizes of ad hoc network in which our supposed network topology structure is multi-cluster classified to form large network.This scheme does not need certificate management, and has simple key distribution process. It can get high security with a few traffic and computation. Besides, it can join the network and leave the network neatly and securely.

The rest of this paper is organized as follows. Section II reviews the previous work related to key management of MANETs. Section III introduces the preliminaries knowledge of bilinear pairing and threshold technique. Section IV presents our key management scheme which includes security parameter initialization and secure communication, key updates, etc. Section V evaluates the performance of the proposal in detail. Finally, section VI presents the conclusion.

2 Related works

Key management schemes of MANETs have been discussed by rich literatures. These schemes can be mainly classified as three types: the Symmetric Key Cryptography scheme, the Asymmetric Key Cryptography scheme and the Group Key Management scheme.

Certificate authorities (CA) which is the vulnerable part in network are usually adopted by the traditional Asymmetric Key Cryptography scheme [2]. As long as CA is not distributed, CA may be infiltrated by adversary with subscriber certificate, personate certificate and removing certificate. CA must ensure availability at any time to perform key management. CA is designed to manage node key's update / rejigger in the system, and issue certificate for the node that is joining the system. In MANET, the credit of single CA can be distributed to a group of nodes which are responsible for key management together. Such a scheme includes: partly distributed authentication scheme and entirely distributed authentication scheme, self-signed certificate, identity-based authentication scheme, and combined public key cryptography scheme.

The Symmetric Key Cryptography scheme [3] can usually be applied to MANET. These schemes are based on key deployed in advance which include single key used by all nodes. Each node shares a single key with another single node or multi-nodes . Each deployed node possesses a following key. These schemes can be divided into two categories: determinate key management scheme and stochastic key management scheme.

Secure group communication of MANET demands the Group Key Management scheme. The donated group key security protocol should solve such problems as group key produce, group key update regularity, group key update when a node member joins in or a node member leaves etc. These group key management schemes can be divided into three basic categories: centralized group key agreement protocols (CGKAP), decentralized group key management protocols with relaying (DeGKMP), and Distributed group key agreement protocols (DiGKAP). CGKAP schemes [4] mainly depend on group controller (GC) to realize the security of group key. The problems in centralized schemes include performance bottleneck, the central point failure and the unrealistic demand for trusted centralized authority in MANET. DeGKMP [5] divides a big group into several subgroups, and each subgroup has a subgroup controller (SC) which is responsible for key management. The unsolved problem is that centralized nodes also exist in DeGKMP. Another problem is that it will produce a lot of communications burden for relaying messages of SC in MANET. DiGKAP [6] schemes generate the group key by uniform distribution from all members , and each member is assumed to have equal work load. The computation cost needed on all members is also uniform, all members generate group key through cooperation. Compare with CGKAP and DeGKMP, DiGKAP is more suitable to be applied in MANETs.

In fact, the first need of MANET security communication is identity authentication which is based on asymmetric key cryptography, and the second need is secret message communication based on symmetric key cryptography. So, a lot of suitable choices for the key management scheme of MANETs are presented with both asymmetric cryptography and symmetric cryptography. In general, the key management scheme of MANET should be realized on line.

Asymmetric Key Cryptography schemes usually depend on certificate based cryptography (CBC), which uses public key certificates to authenticate public key. The CBC-based scheme need certificate based public-key generation and distribution. These schemes don't fit for MANETs because they may cause both unfavorable communication latency and tremendous communication overhead.

In ID-based cryptography, public keys can be derived from entities' known identity information. ID-based cryptography eliminated the need for public key distribution and certificates. So, ID-based cryptography has become a very important tool in research on MANETs' key management.

Several practical identity-based schemes [7, 8, 9] have been presented since 1984. Boneh and Franklin devised the first functional identity-based encryption scheme (IBE) in 2001 [10]. They cleverly used bilinear map (the Weil or Tate pairing) over super-singular elliptic curve [11]. B.Lynn extended the Boneh-Franklin scheme to build an efficient authenticated IBE scheme [12]. Since that time, many other identity-based security schemes for different purposes [13,14,15] have been devised.

IBE schemes require a trusted third party (also called PKG) to be involved on-line in performing the cryptographic operations which can be used to generate and distribute private keys. It's disadvantages include: 1) a bottleneck on the PKG will be created and will become an obstacle for an efficient key management scheme. 2) PKG knows each user's secret, which may cause the problem of key escrow.
The key management service is distributed among certain number of PKGs to improve the IBE scheme. Several parties could share the secret through an (*t*, *n*) threshold scheme, only more than t parties out of n PKGs could cooperate to generate a valid public/private key that executes key management services.

3 Preliminaries

3.1 Bilinear Map and Pairing

The method of bilinear map and pairing should be adopted in practice for efficiency and security. Its basic theory in detail will be discussed.

Suppose p, q be two large primes. Literature[8,9] gave the conditions that p, q must satisfy. E/Fp indicate an elliptic curve $y^2 = x^3 + ax + b$ over the finite field Fp. $G1$ denote a q-order subgroup of the additive group of points of E/Fp, and $G2$ denote a q-order subgroup of the multiplicative group of the finite field Fp. The Discrete Logarithm Problem (DLP) should be hard in both $G1$ and $G2$. For us, a pairing is a map $\hat{e}$: $G1 \times G1 \rightarrow G2$ with the following properties:

1) Bilinear: $\vee$ P, Q, R, S $\in G1$,

$$\hat{e}(P+Q,R+S) = \hat{e}(P,R)\,\hat{e}(P,S)\,\hat{e}(Q,R)\,\hat{e}(Q,S) \quad (1)$$

Consequently, for $\vee$ a, b $\in Z_q^*$, we have

$$\hat{e}(aP, Q)^b = \hat{e}(P, bQ)^a = \hat{e}(P, Q)^{ba} \quad (2)$$

2) Non degenerate: there exists $P,Q \in$ G1 , such that

$$\hat{e}(P,Q) \neq 1. \quad (3)$$

3) Computable: There is an efficient algorithm to

compute $\hat{e}(P,Q)$ for all

$$P,Q \in G1 \quad (4)$$

The Bilinear Diffe-Hellman Problem (BDHP) is hard. Examples of such bilinear maps include modified Weil [16] and Tate [17] pairings in which we can get more comprehensive description of how these pairing parameters should be selected in practice for efficiency and security.

3.2 Secret Sharing Technique

In order to share a secret among n users, we usually adopt secret sharing technique. Later, all the shareholders can get together and recover the secret. We can not recover or reconstruct the secret for less than the needed number of users.

A (*t*,*n*) threshold secret sharing scheme based on polynomial interpolations has been proposed by Shamir in [18]. This scheme is described below:

1) First, we select t points on the plane ($x1$,$y1$), …, (xt, yt), every xi is distinct, there exists an unique polynomial f of degree t-1,such that $f(xi) = yi$ for all i. After getting the t points, one can recover f by using the Lagrange interpolation formula.

2) This also holds in the field Zp, p is prime.

By doing the following steps, we can share a secret S among t parties:

1) Let S be the secret chosen from Zp, p is prime.

2) Select a random polynomial $f(x) = f_0 + f_1x + \ldots + f_{t-1}x^{t-1}$, under the condition that $f_0 = S$ and $f_1, f_2, \ldots, f_{t-1}$ are chosen randomly from Zp.

3) For all i $\in [1,t]$, distribute the shares $Si = (i, f(i))$ to the i-th party.

If the secret has been shared, we can now reconstruct it from every subset of t shares by the Lagrange formula. When given t points (xi, yi), $i =1,\ldots, t$, we get

$$F(x)=\sum_{i=1}^{t} f(x_i)\prod_{i=1,,j\neq i}^{t}(x-x_j)/(x_i-x_j) \bmod p \qquad (5)$$

In a word, Shamir's secret sharing scheme does not depend on the computational power of any party, and it is perfectly secure, flexible and efficient.

4 Our Key Management Scheme

Our three level Key Management Scheme is based on cluster and tree structure. We will discuss it in detail in the following.

4.1 The topology structure of MANET

MANET nodes are first divided as multi-clusters by applying cluster making algorithm. Suppose nodes in each cluster compose the third layer network of the MANETs and cluster head can be selected. All cluster head nodes of the third layer network are also divided as multi-clusters by applying cluster making algorithm and these clusters compose the second layer network of the MANETs. After forming the second layer network, each cluster in the second layer network can select a cluster head and all the cluster heads can also compose the first layer network.

MANETs can be formed with three category nodes: Trusted third part nodes (TTP), cluster head nodes C and ordinary cluster member nodes.

Figure 1 shows the three level key management scheme in MANETs. TTP can only be applied in the phase of MANET initialization which will be discussed in the following paragraph. After initialization process, TTP is in off-line state. Cluster head C is charged with two tasks: 1) It realizes three level distributed KDC, carry through system key management. First, every node in the third layer cluster can produce distributed secrete share by (t,n) threshold cryptography scheme. Each cluster head in the third layer cluster can hold public/private key pair of its own cluster. Second, cluster head in the third layer produce public key with its own identities.It regards the third layer cluster main private key as the second layer cluster private key share which will get together to produce the second layer cluster main private key by (t,n) threshold secret sharing scheme. With the same manner, every cluster head in the second layer may regard its cluster main private key as private key share in the first cluster, and then we can get node public key/private key of the first layer cluster, in the end, we can get the first layer cluster main public key/private key by applying (n,t) threshold secret sharing scheme. 2) Cluster head C has the whole functions of distributing key management for those nodes belonging to cluster head C.

Suppose every node has only one ID (such as network card MAC address) number in the temporary ad hoc network. Different node has different ID number. Each node has the function of finding neighbour node and acquiring its ID information. In the mean time, node has the function of monitoring neighbour node behavior and judging if a neighbour node is a hostile node or a shadiness node.

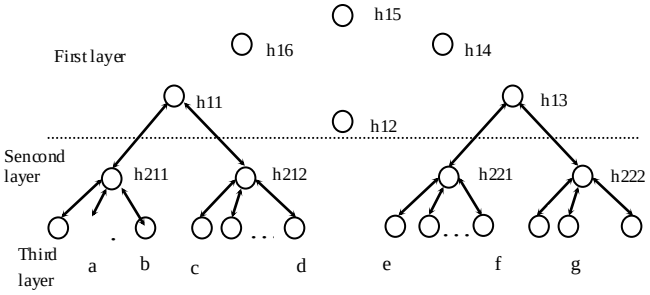

Figure 1. MANETs three level key management scheme

Before MANET is formed, the initial set of participating nodes had got the system parameters which are adopted by the MANET. Each node of this set has the same common parameters. When a new node wants to join the MANET, it will first be authenticated by its neighboring nodes, and then the node and the other nodes belonging a cludter can form a threshold-based Private Key Generation Service (PKG), in which a new master public key is generated for the identity-based cryptosystem and a new master secret key is shared among the nodes. They are in a threshold (t-out-of-n) manner such that fewer than t nodes cannot recover the new master

secret key. Then the join node can obtain corresponding personal private key by having a share of their private keys from each of the *t* nodes that formed the PKG. If we have correct t-shares, corresponding node can compute its personal private key. Personal public key can be got by using a one-way hash function and its own identity.

Anyway, when a new node joins the network, it must first be authenticated by a set of nodes of the MANET and then construct a new master secret key of the MANET by *t* neighboring nodes. In order to resist mobile adversaries' attack, we have evolved shamir's (t,n) threshold secret share scheme in the MANET.

4.2 System Parameters Generation

It is well known that user's public key is computed by using a one-way hash function and its own identity in identity-based public key cryptography. The system parameters can be selected by trusted third party public key generation center (PKGC). The PKGC runs the common parameter generation algorithm A. It works as follows:

1) Suppose a security parameter $k \in Z+$, runs A with input k to generate a prime q (q is large enough to make it infeasible to solve discrete logarithm problem in $G1$ and $G2$), $G1$, $G2$ are two cycle groups of order q, and P is the generator of G_1, and an admissible bilinear map $e\hat{}: G1 \times G1 \rightarrow G2$.

2) Select two hash functions $h_0: \{0, 1\}^* \rightarrow G1$, which maps arbitrary binary strings to nonzero elements in G1, $h_1: \{0, 1\}^* \rightarrow Z_P^*$, which maps arbitrary binary strings to integers in Z_P^*.

3) PKGC chooses stochastic number $S_{PKG} \in Z_P^*$ as its own private key, and the corresponding public key is $Q_{PKG}=S_{PKG}P \in G_1$.

4) Suppose (E_K,D_K) is a pair of symmetric encryption and decryption algorithm. For participator ui (i=1, 2,…,n), its public key is $Q_i=h_0(ID_i)$ and its private key is $S_i=S_{PKG}Q_i$.

5) The system common parameters (P, q, e, h_0, h_1, $G1$,$G2$, S_{PKG}) are pre-distributed to each PKG node through TTP.

4.3 Construct Three Level Key Management Schemes when MANETs are formed

Literature [19] indicates that in three layer network topology structure, if we adopt distributed structre as the first layer network, centralized structure as the second layer network and the third layer network (DCC), it costs less in communication to acquire the same security performance when compared with other network topology structure.

The structure in figure 1 is DCC mode. The first layer network adopts distributed keymanagement in which shamir (t,n) threshold secret sharing scheme is applied to manage public key/private key of cluster main key and each node's key in the cluster.

When the node in the first layer is cluster head and the nodes in second layer are cluster members, we can get the key management methods of the second layer cluster as follows: If a node wants to join or leave the cluster, we adopt ditributed key management mode in order to ensure secure key distribution.

When nodes in the second layer cluster or in the third layer cluster communicate with each other, this cluster topology structure becomes tree structure and the cluster head becomes tree root.The key management mode in the third layer network is the same as the mode in the second layer network.

In the following, we will separately investigate node public key/private key management in distributed key management manner, point to point secure communication channel setup, group key management content, etc. Then we will evaluate this security scheme.

4.4 The formation of The MANET Secure Channel

When a set of nodes are ready to form a MANET, they authenticate and communicate on the master public key and master private key. The node's identity-based public key system is constructed by master private key and node's identity.

Now we will discuss secure channel. If node ui wants to send message m to node uj securely, it will do the following things:

1) It randomly selects $x_j \in Z_P^*$,and calculate

$k_j=(k_{j,1},k_{j,2})=h_1(\hat{e}(Q_j,Q_{PKG})^{xj})$; ui calculates

$r_j= m_j^{kj,2}$, $c_j= E_{kj,1}(m_j||r_j)$, $R_j=r_jQ_i$ and

$s_j=x_jQ_{PKG} - r_jS_i \in G_1$;

2) ui sends (c_i, s_i, R_i) to the receiver uj.

After receiving the message (c_i, s_i, R_i), uj calculates $k_j=(k_{j,1},k_{j,2})=h_1(\hat{e}(Q_j, s_j)\hat{e}(S_i, R_j))$.

Then uj decrypts cj and gets $mj||rj= D_{kj,1}(c_j)$, if $r_j= m_j^{kj,2}$ is correct, uj will think the secret is sent successfully.

4.5 Master Key Form and Update

In the scheme we will improve Shamir's secret sharing. After the MANET is formed, the master public/private key pair should be formed and then be updated through the MANET nodes' cooperation. Each node *Vi* randomly chooses a secret *fi0* and a polynomial $f_i(x) = f_{i0} + f_{i1}x + \ldots + f_{i,t-1}x^{t-1} \pmod q$, and $f_i(0) = f_{i0}$. Through the polynomial, node *Vi* can compute $Sij = fi(j)$ and $Si,i+1 = fi(i+1)$ for $i= 1,2, \ldots, n$ *and* $j= 1,2, \ldots, n$. as his subshare for node *Vj*. V_i sends *Sij* and $S_{i,i+1}$ to *Vj* by a securely channel which has been described above. Node *Vj* may construct its share of master private key from the n receiving subshares. So $K_j=\sum_{i=1}^{n}S_{ij}=\sum_{i=1}^{n}f_i(j)$. So, any coalition of *t* shareholders could jointly recover the secret using $\sum_{i=1}^{t} K_i\, l_i\,(x) \bmod q$ where $li(x)$ is the Lagrange coefficient. It is easy to see that the jointly updated master private key is

$$S_{PKG}(new)=\sum_{i=1}^{n}f_{i0}+\sum_{i=1}^{n}f_i(i+1)=\sum_{i=1}^{n}f_i(0)+\sum_{i=1}^{n}f_i(i+1)$$

$$= \sum_{i=1}^{t} K_i\, l_i\,(0\,) \bmod q + \sum_{i=1}^{n}f_i\,(i+1) \qquad (6)$$

After $S_{PKG}(new)$ is produced above, we will not worry about adversary's attack on (t,n) secret share scheme. Even if the adversary get $\sum_{i=1}^{t} K_i\, l_i\,(0\,)$, we can also use the $S_{PKG}(new)$ securely.

We have chosen P as the generator of G_1 in the phase of System parameters generation. Each shareholder publishes KiP by sharing the master private key. Then the master public key could be computed as $Q_{PKG}\,(new)= S_{PKG}(new)P$.

4.6 Identity-Based Public Key and Private Key Update

If a node in MANET wants to obtain its own public/private key pair, it should contact at least *t* neighbor nodes, present its identity and send the requests to *t* PKGs. All the network nodes share the master private key, and each of these nodes could be the PKG service node. Having Received the request signal, the *t* nodes work together to issue the public/private key pair for the request node. For the identity-based cryptosystem, the public key may be arbitrary string, such as MAC address, e-mail address, etc. In this scheme, we add a time stamp to the public key, to make MANET not in the inconvenient situation when the private key is lost. Once the public key expired, the node's new public key and corresponding private key must be obtained. Public key is constructed as $pk\mathrm{ID} = h_0\,(ID)$, where h_0 is a cryptographic hash function and *ID* is the identity of the requesting node. After acquiring the public key, each of the PKGs service nodes should cooperate to gain the corresponding private key *sk*. Each node of the PKGs computes $s_{ij}=K_i\, h_0\,(ID_j)$ ($i =1,\ldots, m, m>t$), and sends it to the requesting node *Vj*. After collecting all the correct shares of the private keys, the requesting node *Vj* can construct its own private key as $S_{PKGj} = \sum_{i=1}^{m} s_{ij} + x_j$ where x_j is a random number selected by node V_j.

4.7 New Node Join cluster

If a new node V_m wants to join the MANET, it will first randomly choose a secret *fm0* and a polynomial $f_m(x) = f_{m0} + f_{m1}x + \ldots + f_{m,t-1}x^{t-1} \pmod q$, and $f_m(0) = f_{m0}$. Then it will broadcast its public key ID_m to all the nearby m' nodes.

After getting m' ($n \geq m' \geq t-1$) nodes identity in a cluster of the MANET, node V_m and the m' nodes can acquire new cluster public key /shared private key by master key update algorithm described above. Where n is the total number of the cluster nodes. Then the joined node can get new master public key, its share of the new master private key and personal private key.

The other (n-m') nodes in the cluster should update their new master public key / master private key and personal private key as follows. First, any node V_s in the (n-m') nodes will broadcast its public

key *ID* to all the nearby t nodes who possess of new key and request the new master public key, its share of the new master private key and personal private key.The sending message should be *ID*||Request join||$[h_0($ *ID*||Request join $)]_{SPKGs}$ where $[h_0(ID||$Request join $)]_{SPKGs}$ means the message $[h_0(ID||$Request join $)]$ is signed with private key S_{PKGs}

.If the verification process succeeds, the master private key could be computed as follows : for the requesting node *Vs*, each coalition node *Vi* computes the partial share $K\ is = K_i\ l_i(s)$ for node V_s. Here, $l_i(s)$ is the Lagrange term. It sends the partial share K_{is} to node V_s through a secure channel described above.

Node V_s obtains its new share by constructing the partial shares as $K_s = \sum_{i=1}^{t} K_{is}$. And the master public key could be computed as $Q_{PKG}\ (new) = S_{PKG}(new)P$ (P is a common parameter of the system) where $S_{PKG}(new) = \sum_{i=1}^{t} K_i\ l_i\ (0)\ \mathrm{mod}\ q + \sum_{i=1}^{(m'+1)} f_i\ (i+1)$. $\sum_{i=1}^{(m'+1)} f_i\ (i+1)$ can be accepted from a few m' nodes by secure channel introduced above.

In the final step, Each of the PKGs computes $K_{im'} = K_i \bullet h_0(ID_m)$ and sends $K_{im'}$ to the node V_m. After receiving all the correct shares of the private keys, the requesting node V_m could construct its own private key as $S_{PKGm} = \sum_{i=1}^{m} K\ im' + x_m$ where x_m is a random number selected by node V_m .

When a new node join the cluster, all nodes in the cluster only update the cluster master key (public key / private key) in the cluster. And nodes in other cluster will not update their key and the nodes in the cluster will keep old / new key.When it communicates with other cluster nodes, it should adopt old key. And it only adopts new key when communicating with the new node. New node will be monitored by the other nodes of the cluster and it can communicate only in the joined cluster, after some time, the new node can be confirmed to be believable, then the new key will be accepted by other cluster. The monitor algorithm can be found in many literatures.

4.8 Node Leave

If the leaving node is located in the third layer, after it departs from the cluster, the other nodes in the cluster will begin a share key update phase described above. Then the cluster form from the third layer cluster head will update its own cluster key by master key update process. In the end, the first layer node will update its own master key immediately. Key updates take place from lower layer to upper layer.

If the left node is located in the second layer, as soon as it departs from the cluster, the second layer cluster will update its key, then the first layer cluster will update its key like the same action above. But in the third layer, the master key should be updated as follows. First, the node in the second layer composes a cluster with nodes in third layer, so the master key will be the second layer node's key. In order to get the same master key, the second layer node should select the share update polynomial $fu(x) = f_0 + f1x + \ldots + ft\text{-}1x^{t-1}$ (mod q), such that $fu\ (0) = f_0$ to be the second layer node's private key. As the second layer node is also the cluster head of the third cluster, to ensure S_{PKGu} is not changed, in the beginning of share update phase, nodes in the third layer cluster are chosen to collaboratively generate a random share update polynomial $ft(x) = ft1x + \ldots + ft,t\text{-}1x^{t-1}$ (mod q), such that $ft\ (0) = 0$ to ensure S_{PKGu} is not changed. Then each PKG node achieves new secret shares (S1,*new*, S2,*new*,…,S*t*,*new*) and generates its sub-shares (*Si*1,*Si*2,…,*Sit*). Later, when *t* PKG nodes are chosen to work together, they can distribute every sub-shares (*S*1*j*,*S*2*j*,…, *Stj*) to node *Vj* securely, node *Vj* gets the sub-shares (*S*1*j*,*S*2*j*,…, *Stj*) and computes the new share from above shares.

If the left node is located in the first layer, after it departs from the cluster, the first layer cluster will update its key immediately, then the second layer cluster will update its key with the same method as the update from second layer to third layer described above. Then the third layer cluster will do the same action as the second layer cluster does.

And with the help of verifiable secret sharing [20], it can also detect an invalid partial signature.

4.9 Nodes Securely Communicate Across Clusters

When a node wants to communicate with another node in MANET, It will pass through

different passage according to the two nodes' location in MANET. In the following, we will analyse different situations based on figure 1.

If node of a cluster of the third layer (such as node *a*) wants to communicate with a node in the same cluster (such as node *b*), it will send the message to node b directly. If node *a* wants to communicate with a node of another cluster of the third layer (such as node *c)*, it will adopt rout as: node a -> node h211 -> node h212 -> node c . If node *a* wants to send message m to node f, it will take route as follows: node a -> node h211 -> node h11 -> node h13 -> node h221 -> node f. send m to node h11. The principle is that the node who wants to send message first look up target node in the same cluster, if it don't find the target in the same cluster, it will look up the target node in the up or down cluster head node to decide how to forward the message.

5 Performance evaluation

5.1 The Character of Our Scheme

The scheme based on tree and cluster structure adopts many popular security methods. It brings us the following characters:

1) In the forming phase of MANET, we present the secure channel which gives a security algorithm. It is right :

$$\hat{e}(Q_j,s_j)\,\hat{e}(S_i,R_j) = \hat{e}(Q_j,Q_{PKG})^{xj} \qquad (7)$$

Proof:

$$\hat{e}(Q_j, s_j)\,\hat{e}(S_i, R_j)=\hat{e}(Q_j, x_jQ_{PKG}-r_jS_i)\,\hat{e}(S_i, r_jQ_i)$$

$$= \hat{e}(Q_j, x_jQ_{PKG}-r_jS_i)\,\hat{e}(S_i, Q_i)^{rj}$$

$$= \hat{e}(Q_j, x_jQ_{PKG}-r_jS_i)\,\hat{e}(S_{PKG}Q_i, Q_i)^{rj}$$

$$=\hat{e}(Q_j, x_jQ_{PKG}-r_jS_i)\,\hat{e}(Q_i, S_i)^{rj}$$

$$= \hat{e}(Q_j, x_jQ_{PKG}-r_jS_i)\,\hat{e}(Q_i, r_jS_i)$$

$$= \hat{e}(Q_j, x_jQ_{PKG}-r_jS_i + r_jS_i)= \hat{e}(Q_j, x_jQ_{PKG})$$

$$= \hat{e}(Q_j, Q_{PKG})^{xj} \qquad (8)$$

2) We can realize secure communication by secure channel and this make us almost not need memory in symmetric key.
3) This scheme come down to almost every aspect of key management in MANETs, including symmetric key communication, asymmetric communication, identity authentication, etc.

5.2 The Security and Efficiency of The Scheme

The proposed identity-based key management scheme has a low communication overhead and reduced computational consumption when compared with certificate-based key management.

1) It adopts ECC (Elliptic Curve Cryptography), the scheme provides high security level with 160 *bit* keys and is equivalent in strength to RSA with 1024 *bit*. So, based on each node's identity, the public/private key can be much shorter compared with the public key in RSA cryptosystem.
2) In order to ensure node identity is authenticated and secure communication is acquired, we preload the initial security parameters for each node. We don't need any certificate in the scheme.
3) We improve shamir (t,n) threshold secret share scheme as follows: we don't need update sub-share secret regular to resist on adversary, for we introduce a appended random term in the final master secret key.
4) We construct a three level key management scheme which reduce the burden of a big MANETs and add the security of the network. Our scheme has DCC three level structure, so it has less communication overhead.

For example, when compared with the key management scheme in literature [1], our scheme don't need many memor-ies to save keys for symmetric key encryption communication. When compared with the scheme in literature [21], our scheme overcome the obvious deficiency that it has no identity authentication when two nodes communicate with each other.

In a word, the ID-based three level key management scheme should be an efficient and

security alternative when compared with the traditional certificate-based method.

6 Conclusion and Future Work

In summary, the proposed scheme is based on identity-based and threshold three level key management scheme which adopts Elliptic Curve Cryptography (ECC) in this paper. It is well known that ECC is appropriate for such nodes due to its smaller keys and its higher security levels. (*t, n*) threshold secret sharing algorithm has been improved to resist adversary's attack and ensure availability of network services, such as key generation and key distribution. The pairing technology provides authentication and confidentiality with reduced communication overhead and computational cost. Our scheme which combined the identity-based and threshold cryptography can satisfy the secure requirements of MANET.

In the future, we will simulate our scheme in detail. Based the simulation results, we will improve our scheme in order to get the most efficient scheme. In the mean time, we will find some better methods to realize ad hoc network's security .So, we have enough ability to acquire a secure, efficient and applied key management scheme.

References:

[1] HU Rong-lei, LIU Jian-wei, ZHANG Qi-shan. "Cluster-based key management scheme for ad hoc networks". Journal on Communications, china. Vol.29 No.10. October 2008. pp.223-228.

[2] H. Luo, J. Kong, P. Zerfos, S. Lu, and L. Zhang, "URSA: Ubiquitous and Robust Access Control for Mobile Ad Hoc Networks, " IEEE/ACM Transactions on Networking , December 2004, pp. 1049-1063

[3] W. Du, J. Deng, Y. S. Han, P. Varshney, J. Katz, and A. Khalili, "A Patrwise Key Predistribution Scheme for Wireless Sensor Networks," ACM Transactions on Information and System Security (TISSEC), 2005 , pp. 228-258.

[4] D. Wallner, E. Harder, and R. Agee, "Key Management for Multicast: Issues and Architectures", RFC 2627, 1999.

[5] A. Ballardie, "Scalable Multicast Key Distribution", RFC1949, 1996.

[6] N. Asokan and P. Ginzboorg, "Key Agreement in ad hoc networks", Elsevier Journal of Computer Communications. Vol. 23, pp. 1627 – 1637, 2000.

[7] A.Shamir: "Identity based cryptosystems and signature schemes." *Advances in Cryptology – Proceedings of Crypto'84* pp. 47-53, 1984.

[8] A.Fiat, A.Shamir: How to Prove Yourself: practical Solutions to Identification and Signature Problems.

[9] L.Guilou, J-J. Quisquater: A "Paradoxical" identity based Signature Scheme Resulting From Zero-Knowledge. *Advances in Cryptology – Crypto'88, LNCS 0403, Springer* (1988) 216-231.

[10] D. Boneh and M.Franklin: Identity-based Encryption from the Weil Pairing. *Proc CRYPTO'01* (2001) 213-229.

[11] D. Boneh and M.Franklin: Identity-based Encryption from the Weil Pairing. *SIAMJ. Computing*, vol. 32, no. 3(2003) 586-615.

[12] B.Lynn: Authenticated Identity-based Encryption. available at *http://eprint.iacr.org* (2002).

[13] WATERS B R: Efficient identity-base encryption without random oracles[R]. Cryptology ePrint Archive, Report 2004/180, *http://epring.iacr.org* (2004).

[14] Javier Herranz: Identity-based ring signatures from RSA. *Theoretical Computer Science* 389(2007) 100-117.

[15] J.Pan, L.Cai, X.Shen, J.W.Mark: Identity-based secure collaboration in wireless Ad hoc networks. *Computer Network* 51(3) (2007) 853-865.

[16] G.Frey, M.Muller, H.riick: The Tate pairing and the discrete logarithm applied to elliptic curve.

[17] A.Menezes, T.Okamoto, S.VanstoneL: Reducing elliptic curve logarithms to logarithms in a finite field.

[18] Shamir.A: How to share a secret. *Communications of the ACM* vol.22 no.11 (1979) 612-613.

[19] Bin Sun; Bin Yu; The three-layered group key management architecture for MANET.Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on Volume 02, 15-18 Feb. 2009 Page(s):1378 – 1381

[20] T.Pedersen: Non-interactive and informationtheorectic secure verifiable secret sharing. In J.Feigenbaum, editor, Afvances in Cryptology- Crypto'91, *the 11th Annual International Cryptology Conference, Santa Barbara*, CA USA, August 11-15,1991,

Proceedings, Volume 576 of Lecture Notes in Computer Science, Springer(1992) 129-140.

[21] Yuchen Zhang; Jing Liu; Yadi Wang; Jihong Han; Hengjun Wang; Kun Wang.” Identity-Based Threshold Key Management for Ad Hoc Networks” Computational Intelligence and Industrial Application, 2008. PACIIA '08. Pacific-Asia Workshop on Volume 2, 19-20 Dec. 2008 Page(s):797 - 801