A Verifiable Dynamic Threshold Key Management Scheme Based on Bilinear Pairing without a Trusted Party in Mobile Ad Hoc Network*

Xianyong Meng\(^{1,2}\)
\(1\)Department of Electromechanical Engineering
University of Macau
Av. Padre Tomas Pereira, Taipa, Macao, China
\(2\)Zhuhai College of Jilin University
Zhuhai 519041, China
meng_xianyong@hotmail.com

Yangmin Li\(^{1,3}\)
\(3\)School of Mechanical Engineering
Tianjin University of Technology
Tianjin 300191, China
Corresponding author: ymli@umac.mo

Abstract—A dynamic threshold key management scheme based on bilinear pairing without a trusted party is proposed in this paper. The system can dynamically add, modify and remove a member in MANET, and can change the threshold value according to the scale of the group. Adopting distributed framework generates the system key, which is constructed by all participants collectively, it is not like most presented schemes generated by Key Generation Center (KGC). In our threshold scheme, any participant can recover the system public key, but can not reconstruct the system secret key without any other \(t-1\) participants’ help. The designated combiner (DC) can verify the correctness of shared secret during the recovery phase of system key. Simultaneously, the threshold polynomial based on elliptic curve cryptography (ECC) is obviously more secure and efficient than the previous schemes’ polynomial based on prime-field. Additionally, we design a new feasible bilinear pairing based signcryption scheme, which is efficient in terms of both computational complexity and communication load and can realize security communication between any two participants. The security analysis shows that the proposed scheme is more feasible, secure and efficient compared with the previous schemes on Mobile Ad Hoc Network.

Index Terms—Threshold Key Management, Mobile Ad Hoc Network, Bilinear Pairing, Signcryption, Without a Trusted Party.

I. INTRODUCTION

The Mobile Ad Hoc Network (MANET) is a temporary wireless network consisting of a set of autonomous, mobile and non-infrastructure nodes, which is used to establish a peer-to-peer network in special occasion where environmental constraints demand totally distributed network [1], such as in the application field of battlefield, disaster relief and interactive conference, etc.

Due to lacking of any pre-existing infrastructure and management center in MANET like basic station, router and switcher in conventional network, which can carry out basic networking functions [2], MANET is composed of resource-constrained devices with the limited calculation capacity, communication ability and storage space, etc. However, due to the inherent characteristics of MANET, such as connecting over a shared wireless medium, lacking of management center and vulnerability against adversarial attacks [3], there are more and more security problems of distributed communication to be solved in MANET compared with the mature trusted center network [4]. As a viable solution for MANET, dynamic threshold key management scheme without a trusted party for decentralized network has become a research focus recently, because the system keys of MANET are constructed by all the participants without a trusted party, which can work as a trusted center to generate and management the keys.

The concept of \((t, n)\) threshold key management scheme was proposed by Shamir in 1979 firstly [5]. In \((t, n)\) threshold key management scheme, any \(t\) or more than \(t\) shareholders can collectively recover the secret, but any set of less than \(t\) shareholders can not regenerate private key of the system. Based on Shamir’s scheme, some threshold key management schemes [6], [7] are proposed and applied to centralized network depending on a CA known as a trusted party. However, It is difficult to construct a credible CA within the self-organization and peer-to-peer MANET where there are no pre-existing control center and infrastructure. In such a distributive environment, only threshold key management scheme without trusted party is perfectly suitable. There have been some literature [8], [9] on threshold key management schemes without a trusted party in MANET. These threshold schemes define a \((t-1)\)th degree polynomial based on a prime-field with a relatively weak encryption intensity compared to the \((t-1)\)th degree polynomial based on elliptic curve cryptography (ECC) or bilinear pairing [10]. In order
to provide equivalent security, the \((t - 1)\)th degree polynomial based on a prime-field must enhance the coefficient length of polynomial. So, it will increase occupancy rate of resources and computational complexity with limited resources. In this paper, we proposed a novel \((t, n)\) threshold key management scheme, which constructs a \((t - 1)\)th degree polynomial based on ECC. Because of the high encryption intensity of ECC, the ECC is quite feasible for the constrained environment. We use a serial x-coordinate [11] of points on elliptic curve as the coefficients of \((t - 1)\)th degree polynomial of \((t, n)\) threshold key management scheme. So, the polynomial of new \((t, n)\) threshold scheme based on ECC is obviously more safe and simple than those polynomials based on prime-field of schemes [8], [9].

In our scheme, the system keys are generated by \(t\) or more than \(t\) participants collectively, rather than those of the previous threshold key management are generated by Key Generation Center (KGC). Firstly, each participant randomly constructs a \((t - 1)\)th degree polynomial based on ECC respectively, then all the participants must collaborate to construct the initial \((t - 1)\)th degree polynomial for MANET. Moreover, our scheme is a verifiable \((t, n)\) threshold key management scheme that any participant can verify the shared-secret \(\lambda_{ij}\) which is provided by all the other participants. Meanwhile, the new scheme is a dynamic threshold key management scheme, which is constructed based on bilinear pairing without a trusted party. It means that the system can dynamically add, modify and remove a member in MANET, and can change the threshold value according to the scale of the MANET.

In order to realize secure communication between any two participants in MANET, we design a more simple and efficient signcryption scheme based on the new \((t, n)\) threshold key management scheme without a trusted party. Signcryption is a new cryptographic primitive that performs signature and encryption simultaneously. Compared with the scheme of signature-then-encryption, the signcryption scheme consumes fewer system resources, such as computational costs and communication overheads. Therefore, any two participants can communicate with each other through using the new signcryption scheme based on bilinear pairing on ECC consuming fewer wireless bandwidth, communication cost and storage space, etc. The proposed scheme based on ECC not only satisfies the flexibility of the \((t, n)\) threshold key management scheme, but also possesses the high-efficiency of the signcryption scheme based on bilinear pairing. Consequently, the new scheme provides an efficient solution for security of MANET without pre-existing infrastructure.

The structure of the paper is organized as follows: The concept and some mathematical problems of bilinear pairing will be briefly introduced in Section 2. In Section 3, the \((t, n)\) threshold key management scheme without trusted party is described in detail. The signcryption scheme based on bilinear pairing is designed in Section 4. In Section 5, secure analysis of the scheme is discussed. Finally, the conclusion is drawn in the final part of this paper.

## II. Preliminaries

### A. Bilinear Pairing

We firstly introduce the concept and characteristics of bilinear pairing: Let \(G_1\) be a cyclic additive group generated by \(P\), and \(G_2\) be a cyclic multiplicative group. The \(G_1\) and \(G_2\) have the same order \(q\). Let \(a, b\) be arbitrary numbers, which belong to \(Z_q^*\). A bilinear pairing is a mapping built on \((G_1, G_2)\), let \(\hat{e} : G_1 \times G_1 \rightarrow G_2\) be a map with the following properties [12][13]:

1) **Bilinearity**: For any \(a, b \in Z_q^*\) and \(P, Q \in G_1\), such that \(\hat{e}(aP, bQ) = \hat{e}(abP, Q) = \hat{e}(P, abQ) = \hat{e}(P, Q)^{ab}\);

2) **Non-degenerate**: For any \(P, Q \in G_1\), such that \(\hat{e}(P, Q) \neq I\), where \(I\) is the identity element of \(G_2\).

3) **Computability**: There is an effective polynomial time algorithm to compute \(\hat{e}(P, Q) \in G_2\), for any \(P, Q \in G_1\).

### B. Some Mathematical Problems

Some mathematical problems on bilinear pairing is described as follows:

1) **The Elliptic Curve Discrete Logarithm Problem (ECDLP)**: For any \(a \in Z_q^*\) and \(P, Q \in G_1\), to find an integer \(a \in Z_q^*\), such that \(Q = aP\). We assume that ECDLP in \(G_1\) is hard.

2) **The Computational Diffie-Hellman Problem (CDHP)**: For any \(a, b \in Z_q^*\) and \(P \in G_1\), given \(P, aP, bP\), to compute \(abP\) is difficult.

3) **The Decision Diffie-Hellman Problem (DDHP)**: For any \(a, b \in Z_q^*\) and \(P \in G_1\), given \(P, aP, bP, cP\), decide whether or not \(c \equiv ab \pmod q\) is hard.

4) **The Bilinear Diffie-Hellman Problem (BDHP)**: For all \(a, b, c \in Z_q^*\), given \(P, aP, bP, cP \in G_1\), \(\hat{e} : G_1 \times G_1 \rightarrow G_2\) is a bilinear mapping, the problem of computing \(\hat{e}(P, P)^{abc} \in G_2\) is hard.

In this paper, we assume that the ECDLP and CDHP are hard, it means that there is no an efficient algorithm in polynomial time to work out ECDLP and CDHP with significant probability. The group is a Gap Diffie-Hellman (GDH) group that there exists an efficient algorithm that can solve the DDHP but there is no a valid algorithm for solving the ECDLP or CDHP with non-negligible probability. The GDH group on bilinear pairing can be constructed on supersingular elliptic curves or hyperelliptic curves over finite field.
III. THRESHOLD KEY MANAGEMENT SCHEME

A. Initialization

Let $G_1$ be a cyclic additive group, $G_2$ be a cyclic multiplicative group, two groups have the same order $q$, which is a large prime number. We construct the bilinear pairing: $\hat{e}: G_1 \times G_1 \rightarrow G_2$, the randomly select generator $G$ of order $q$ is a base point of elliptic curve $E$.

We choose three secure one-way hash functions $H_0, H_1, H_2$, which satisfy the following requirements: $H_0 : \{0,1\}^n \rightarrow Z_q^*$, $H_1 : G_1 \times \{0,1\}^n \rightarrow Z_q^*$, $H_2 : G_2 \rightarrow \{0,1\}^n$. Then, the system publishes parameters as $\{G, G_1, G_2, \hat{e}, H_0, H_1, H_2\}$.

B. Generation of Threshold Scheme

In this section, we design a verifiable $(t, n)$ threshold key management scheme based on ECC without a trusted party.

1) Generating the Threshold Polynomial:

The master keys of the new threshold scheme are generated by at least $t$ participants of the system collectively, while those of previous threshold schemes are generated by KGC in MANET.

Each participant $U_i (i = 1, 2, \ldots, n)$ randomly generates a $(t-1)$th degree polynomial: $f_i(x) = F_{0i} + F_{1i}x + \cdots + F_{t-1}x^{t-1} \mod q$. The participant $U_i$ can randomly select $t$ integers $a_{ik} \in Z_q^*(k = 0, 1, \ldots, t-1)$ and calculates $F_{0i} = (a_{0i}G)x$, $F_{1i} = (a_{1i}G)x$, $\cdots$, $F_{t-1} = (a_{t-1}G)x$, where $F_{0i}$, $F_{1i}$, $\cdots$, $F_{t-1}$ are saved by the participant $U_i$ as the secrets of threshold polynomial. In formula $F_{0i} = (a_{0i}G)x$, $(a_{0i}G)x$ is an $x$-coordinate of point $(a_{0i}G)$ on the elliptic curve $E$ [11], and $G$ is a base point (generator of $G_1$) of the elliptic curve $E$.

Each participant $U_i$ randomly selects $s_i \in Z_q^*$ and computes $s_i a_{ik} G(i = 0, 1, \ldots, t-1)$, and then computes $S_i = s_i G(i = 1, 2, \ldots, n)$. Participant $U_i$ publishes $S_i, s_i a_{ik} G$ and keep the secret $s_i, a_{ik}$ The $s_i$ and the $S_i$ are saved respectively as the secret key and the public key of participant $U_i$.

The positive number $ID_i$ represents the identity of the participant $U_i$, such as IP address, E-mail address. For another participant $U_j(j \neq i)$ in the MANET, $U_i$ computes $\lambda_{ij} = f_i(ID_j) \mod q$, and then sends the computing result to $U_j$. The accepter $U_j$ verifies the information $\lambda_{ij} = f_i(ID_j) \mod q$, which is received from different $n - 1$ participants. The participant $U_j$ verifies the following equation:

$$\hat{e}(f_i(ID_j), S_i) = \hat{e}(\sum_{k=0}^{t-1} s_i a_{ik} G(ID_j)^k, G) \quad (1)$$

If and only if the above equation holds, the participant $U_j$ accepts $f_i(ID_j)$, and then computes $\lambda_j = \sum_{i=1}^{n} f_i(ID_j) \mod q$, in which $\lambda_j$ is saved by each participant $U_j$ as secret information.

2) Generating the Master Keys:

The system constructs a $(t-1)$th degree polynomial

$$f(x) = \sum_{i=1}^{n} f_i(x) \mod q = T_0 + T_1x + \cdots + T_{t-1}x^{t-1} \mod q,$$

where the coefficients $a_i \in Z_q^*(i = 0, 1, \ldots, t-1)$ are secret to all participants $U_i$. If we describe the above formula in the form of x-coordinate of point on the elliptic curve $E$, the formula can be expressed as $f(x) = \sum_{i=1}^{n} f_i(x) \mod q = (a_{0i}G)x + (a_{1i}G)x + \cdots + (a_{t-1}G)x \mod q$, where $(a_{t-1}G)x$ is the $x$-coordinate of the point $(a_{t-1}G)$ on the elliptic curve $E$. The coefficient of the $(t-1)'th$ degree polynomial can be computed through the equation $a_0 G = \sum_{i=1}^{n} a_{0i} G$, $a_1 G = \sum_{i=1}^{n} a_{1i} G, \ldots, a_{t-1} G = \sum_{i=1}^{n} a_{t-1 i} G$, any participant cannot recover the coefficient $a_i \in Z_q^*$ without any other $n - 1$ participants' help.

Any participant can obtain $s_i a_{0i} G$ from notice board, and compute the system public key $Y = \sum_{i=1}^{n} s_i a_{0i} G \mod q$, while any participant cannot generate the system secret key $X = \sum_{i=1}^{n} a_{0i} G \mod q$, unless the participant $U_i$ can get helps from any other $t-1$ participants. However, any participant $U_i$ can regenerate the secret key $X$ with at least $t - 1$ participants' help. The relation between $Y$ and $S_i$ can be represented using the formula $Y = \sum_{i=1}^{n} a_{0i} S_i G \mod q$.

Any participant $U_i$ obtains $t - 1$ secret values $\lambda_{j}(j = 1, 2, \cdots, n, j \neq i)$ from $U_j$ firstly, then calculates the secret key $X$ using the Lagrange Interpolation Formula as follows:

$$f(x) = \{\sum_{i=1}^{t} f(ID_i) \prod_{j=1, j \neq i}^{t} \frac{x - ID_j}{ID_i - ID_j}\} \mod q \quad (2)$$

As $\lambda_i = \sum_{j=1}^{n} f_j(ID_j) \mod q$ and $\lambda_i = f(ID_i)$, we can get $f(x) = \{\sum_{i=1}^{t} \lambda_i \prod_{j=1, j \neq i}^{t} \frac{x - ID_j}{ID_i - ID_j}\} \mod q$. In the $(t, n)$ threshold scheme, any $t$ participants can recover the shared secret $X$ using Lagrange Interpolation Formula. After getting $t - 1$ shared secrets $\lambda_j$, the participant $U_i$ reconstructs the shared secret $X = \sum_{i=1}^{n} a_{0i} G = f(0) = \sum_{i=1}^{t} f(ID_i) C_i \mod q$, where $C_i = \sum_{j=1, j \neq i}^{t} \frac{-ID_j}{ID_i - ID_j}$. 

IV. DYNAMIC TOPOLOGY SCHEME

In this section, we describe how to add and remove a participant in dynamic topology network. If a new participant wishes to join the MANET, it must obtain at least $t$ or more participant's approving admission from current MANET. When the new participant becomes a legitimate member of the MANET, it will possess a shared secret like other
participant and construct a \((t-1)\)th degree polynomial. When a participant remove from the MANET, it must notice all the other participants, and then remove the shared secret from the MANET. The dynamic topology protocol includes following steps:

A. Add a New Number

When a new participant \(U_k\) wants to join the MANET, it randomly constructs a \((t-1)\)th degree polynomial: \(f_k(x) = F_{k0} + F_{k1}x + \cdots + F_{kt-1}x^{t-1} \mod q\).

Similar to the initial phase, the participant \(U_k\) can randomly select \(t\) integers \(a_{ki} \in \mathbb{Z}_q^* (i = 0, 1, \cdots, t-1)\) and compute \(F_{k0} = (a_{k0}G)x, F_{k1} = (a_{k1}G)_x, \cdots, F_{kt-1} = (a_{kt-1}G)_x\), where \(F_{k0}, F_{k1}, \cdots, F_{kt-1}\) are the secrets of threshold polynomial. In the formula \(F_{k0} = (a_{k0}G)_x, (a_{k0}G)_x\) is an \(x\)-coordinate of point \((a_{k0}G)\) on the elliptic curve \(E\).

The participant \(U_k\) selects a random number \(s_k \in \mathbb{Z}_q^*\) and computes \(s_k a_{ij} G (j = 0, 1, \cdots, t-1)\) and \(S_k = s_k G\), where \(S_k, S_k a_{ij} G\) is public and \(s_k, a_{ij}\) is secret. The \(S_k\) is secret key and the \(S_k\) is public key of participant \(U_k\), respectively.

The participant \(U_k\) broadcasts identity ID\(_k\) to all the participants. Every other participant \(U_i\) in the MANET computes \(\lambda_k = f_i(ID_k) \mod q\) for \(U_k\), and sends the computing result to \(U_k\). The information \(\lambda_k = f_i(ID_k) \mod q\) received from different \(n\) participants can be validated by the following equation:

\[
\hat{\lambda}(f_i(ID_k), S_i) = \hat{\lambda}(\sum_{j=0}^{t-1} s_i a_{ij} G(ID_k)^j, G) \tag{3}
\]

If validation passes, the \(U_k\) accepts \(f_i(ID_k)\) and computes \(\lambda_k = \sum_{i=1}^{n+1} f_i(ID_k) \mod q\), and then saves it as secret information.

After adding a new participant \(U_k\), any participant \(U_i\) can calculate system public key \(Y = \sum_{i=1}^{n+1} s_i a_{i0} G \mod q\).

But any participant can not know the system secret key \(X = \sum_{i=1}^{n+1} a_{i0} G \mod q\), which can only be regenerated by at least \(t\) participants by using Lagrange Interpolation Formula as follows:

\[
X = f(0) = \left\{ \sum_{i=1}^{t} \lambda_i \prod_{j=1,j\neq i}^{t} \frac{-ID_j}{ID_i-ID_j} \right\} \mod q \tag{4}
\]

B. Delete a Number

When any participant \(U_k\) deviates from the MANET, it broadcasts its identification ID\(_k\) to all the participants. Each participant \(U_j\) recomputes its secret sharing \(\lambda_j = \sum_{i=1,i\neq k}^{n} f_i(ID_j), \) then saves the new result \(\lambda_j\).

Similarly, any participant \(U_j\) calculates the system public key \(Y = \sum_{i=1,i\neq k}^{n} s_i a_{i0} G \mod q\). Any \(t\) participants can compute the system secret key \(X\) using Lagrange Interpolation Formula (4).

C. Dynamic Threshold

The system can adjust threshold value of the key management scheme from \(t\) to \(t'\), and each participant \(U_i\) randomly regenerates the \((t' - 1)\)th degree polynomial \(f_i(x) = F_{i0} + F_{i1}x + \cdots + F_{it'-1}x^{t'-1} \mod q\).

Similarly, the \(U_i\) can randomly choose \(t'\) integers \(b_{ik} \in \mathbb{Z}_q^*(k = 0, 1, \cdots, t' - 1)\) and compute the secret coefficients \(F_{i0} = (b_{i0}G)_x, F_{i1} = (b_{i1}G)_x, \cdots, F_{it'-1} = (b_{it'-1}G)_x\), where \((b_{i0}G)_x\) is the \(x\)-coordinate of point \((b_{i0}G)\) on the elliptic curve \(E\). Each participant \(U_j\) selects a random number \(w_i \in \mathbb{Z}_q^*\) and computes \(w_i b_{ik} G (k = 0, 1, \cdots, t' - 1)\) and \(W_i = w_i G (i = 1, 2, \cdots, n)\), where \(W_i, w_i b_{ik} G\) is public and \(w_i, b_{ik}\) is secret. The \(w_i\) is secret key and the \(W_i\) is public key of \(U_i\), respectively.

The \(U_i\) recompute \(\lambda_{ij} = f_i(ID_j) \mod q\) and sends it to \(U_j\). The participants \(U_j\) validates the information \(\lambda_{ij} = f_i(ID_j) \mod q\) received from different \(n - 1\) participants using the following equation:

\[
\hat{\lambda}(f_i(ID_j), W_i) = \hat{\lambda}(\sum_{k=0}^{t'-1} w_i b_{ik} G(ID_j)^k, G) \tag{5}
\]

If validation passes, the \(U_j\) accepts the value \(f_i(ID_j)\) and saves the value \(\lambda_j = \sum_{i=1}^{n} f_i(ID_j) \mod q\) as secret information.

The system generates a \((t' - 1)\)th degree polynomial \(f(x) = \sum_{i=1}^{n} f_i(x) \mod q = T_0 + T_1 x + \cdots + T_{t'-1} x^{t'-1} \mod q\), where all coefficients \(T_0 = (a_{00}G)_x, T_1 = (a_{10}G)_x, \cdots, T_{t'-1} = (a_{t'-1}G)_x\) are secret to all participants.

Every participant \(U_i\) can calculate system public key \(Y = \sum_{i=1}^{n} w_i b_{i0} G \mod q\), but at least \(t'\) participants collectively generate the system secret key using the Lagrange Interpolation Formula:

\[
X = f(0) = \left\{ \sum_{i=1}^{t'} \lambda_i \prod_{j=1,j\neq i}^{t'} \frac{-ID_j}{ID_i-ID_j} \right\} \mod q \tag{6}
\]

V. Signcryption Scheme

In this section, we design a novel signcryption scheme based on our threshold key management scheme without a trusted party. The signcryption can realize secure communication between any two participants of the MANET. The procedure of the signcryption scheme contains two phases: the signcryption phase, the verification and message recovery phase.
A. Signcryption Phase

The participant $U_i$ signcrypts message $m$ and then sends it to participant $U_j$. In this phase, encryption and signature can simultaneously be fulfilled by sender $U_i$ in the following steps:

- $S_i$ is the public key of the sender $U_i$, $S_j$ is the public key of the accepter $U_j$, $Y$ is the public key of the system, $s_i$ is the secret key of the sender $U_i$. The $U_i$ randomly chooses $k \in Z_q^*$ and computes as follows:
  
  $N = kG$
  
  $S = s_iH_0(m)N + kH_1(N\|ID_i\|ID_j)Y$
  
  $w = \hat{e}(G, s_iS_i)^k$
  
  $c = H_2(w) \oplus m$
  
  $\sigma = \{c, N, S\}$

Then the sender $U_i$ sends the signcryption $\sigma = \{c, N, S\}$ to the receiver $U_j$.

B. Verification and Message Recovery Phase

After receiving the signcryption $\sigma = \{c, N, S\}$, the accepter $U_j$ can recover the message $m$ and verify the signature by the following steps:

- Since $s_iS_j = s_jS_i$, the verifier $U_j$ can compute $w$: $w = \hat{e}(N, s_iS_i)$
- The verifier $U_j$ recovers the message $m$ by the equation: $m = c \oplus H_2(w)$
- Then the verifier $U_j$ verifies the formula:
  
  $\hat{e}(G, S) = \hat{e}(S_i, N)H_0(m)\hat{e}(N, Y)H_1(N\|ID_i\|ID_j)$

The accepter $U_j$ will accept the message $m$ if and only if the above formula holds.

VI. Security Analysis

In this section, we analyze the performances of the threshold key management scheme and relative signcryption scheme, including the correctness, security, and efficiency.

**Theorem 1**: In the initialization phase of threshold key management scheme, the participant $U_j$ can validate the information $\lambda_{ij} = f_i(ID_j)$ received from different $t-1$ participants.

**Proof**: The participant $U_j$ validates the information using the following equation:

$$\hat{e}(f_i(ID_j), S_i) = \hat{e}(\sum_{k=0}^{t-1} s_ia_{ik}G(ID_j)^k, G)$$

As $f_i(x) = F_{i0} + F_{i1}x + \cdots + F_{i(t-1)}x^{t-1} \mod q$, we can get $f_i(ID_j) = a_{0}G + a_{1}G(ID_j) + \cdots + a_{t-1}G(ID_j)^{t-1} \mod q$. Therefore, $\hat{e}(f_i(ID_j), S_i)$ can be calculated as:

$$\hat{e}(\sum_{k=0}^{t-1} a_{ik}G(ID_j)^k, s_iG)$$

$$= \hat{e}(\sum_{k=0}^{t-1} s_ia_{ik}G(ID_j)^k, G)$$

If the equation $\hat{e}(f_i(ID_j), S_i) = \hat{e}(\sum_{k=0}^{t-1} s_ia_{ik}G(ID_j)^k, G)$ holds, the $U_j$ accepts the $f_i(ID_j)$.

**Theorem 2**: In the message recovery and verification phase of the signcryption scheme, the accepter $U_j$ accepts the signcryption if and only if the equations $\hat{e}(G, S) = \hat{e}(S_i, N)H_0(m)\hat{e}(N, Y)H_1(N\|ID_i\|ID_j)$ holds.

**Proof**: Since $s_iS_j = s_jS_i$, the verifier can compute $c = \hat{e}(G, s_iS_i)^k$. Then the verifier $U_j$ can verify the signcryption by the equation $\hat{e}(G, S) = \hat{e}(S_i, N)H_0(m)\hat{e}(N, Y)H_1(N\|ID_i\|ID_j)$.

**Theorem 3**: Our scheme is unforgeable.

**Proof**: Any $m(2 \leq m \leq t)$ participants can not forge the master keys $Y$ and $X$ in MANET using their private sharing secrets $\lambda_i$, because they still face with solving the Threshold Problem.

We assume that the dishonest participant $U_i$ wants to fake the signcryption $\sigma = \{c, N, S\}$. As we know, the dishonest participant $U_i$ must forge $N = kG$, $S = s_iH_0(m)N + kH_1(N\|ID_i\|ID_j)Y$ and $w = \hat{e}(G, s_iS_i)^k$ before it forges $c$, but it is impossible to the adversary to forge $N$, $S$ and $w$ on the condition that the adversary can not obtain the secret keys of the sender $U_i$ and the accepter $U_j$, because it faces with solving the Bilinear Diffie-Hellman Problem.

**Theorem 4**: We propose a $(t, n)$ threshold key management scheme without a trusted party in MANET.

**Proof**: In our scheme, the system keys are generated by $t$ or more participants collectively, rather than those of the previous threshold key management are generated by KG. Firstly, each participant randomly constructs a $(t-1)$th degree polynomial $f_i(x) = F_{i0} + F_{i1}x + \cdots + F_{i(t-1)}x^{t-1} \mod q$ based on ECC respectively, then all the participants must collaborate to construct the $(t-1)$th degree polynomial $f(x) = \sum_{i=1}^{n} f_i(x) \mod q = T_0 + T_1x + \cdots + T_{t-1}x^{t-1} \mod q$ for MANET. Moreover, all participants must attend the construction of the $(t-1)$th degree polynomial of the MANET. Any participant can recover the public key of the MANET, while the secret key of the system can only be recovered by at least $t$ participants.

**Theorem 5**: Our threshold key management scheme and signcryption scheme is efficient.

**Proof**: In this paper, we propose a novel efficient $(t, n)$ threshold key management scheme, which constructs a $(t-1)$th degree polynomial based on ECC with the relatively high encryption.
intensity. Compared to \((t-1)\)th degree polynomial based on a prime-field, the polynomial based on ECC is quite feasible for the constrained environment and has some advantages, such as short key sizes, high encryption intensity, good flexibility, and so on. We use a serial x-coordinate of points on elliptic curve as the coefficients of \((t-1)\)th degree polynomial of \((t, n)\) threshold key management scheme. So, the polynomial of new \((t, n)\) threshold scheme based on ECC is obviously more safe and simple than those polynomials based on prime-field of schemes \([8], [9]\).

VII. DISCUSSIONS ON APPLICATIONS

The MANET can be applied in security communication of multi-robot system and swarm robot system \([14]-[17]\), where all the members of group can collaborate with each other to implement a mission in some open and constrained environment, such as battlefield, disaster rescue and emergency.

In decentralized security communication environment, all the robots collectively generate the master keys of the MANET for multi-robot system. The dynamic \((t, n)\) threshold key management scheme can provide a robust security communication in such a dynamic topology network, where any robot can enter into and depart from the MANET at any time. In a multi-robot system, any two robots can realize security communication using the signcryption scheme, which performs signature and encryption simultaneously. There are many important security issues need to resolve in MANET for multi-robot system, such as the decentralized threshold signature, traceability and routing protocols, and so on \([18], [19]\).

VIII. CONCLUSION

To satisfy security requirements and to improve the security efficiency and feasibility of self-organization and non-infrastructure MANET, we proposed a novel dynamic \((t, n)\) threshold key management scheme based on bilinear pairing without a trusted party. The system can dynamically add, modify and remove a member in MANET, and can change the threshold value according to the scale of the group. We adopt distributed framework to generate the system key, which is constructed by all participants collectively. In our threshold scheme, any participant can recover the system public key, but can not reconstruct the system secret key without any other \(t-1\) participants’ help. The designated combiner can verify the correctness of shared-secret during the recovery phase of the system keys. Simultaneously, the threshold polynomial based on ECC is obviously more secure and efficient than the previous schemes’ polynomial based on prime-field.

Furthermore, we design a new efficient signcryption scheme based on bilinear pairing, which can realize security communication between any two participants. The security analysis shows that the novel scheme is more efficient, secure and feasible compared with the presented schemes on Mobile Ad Hoc Network.

REFERENCES