IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON COLLABORATION FOR INTERNET OF THINGS

Received January 15, 2019, accepted February 12, 2019, date of publication March 6, 2019, date of current version March 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2902226

TAW: Cost-Effective Threshold Authentication With Weights for Internet of Things

ZHENHU NING[1], GUANGQUAN XU[2], (Member, IEEE), NAIXUE XIONG[2], YONGLI YANG[1], CHANGXIANG SHEN[1], EMMANOUIL PANAOUSIS[3], HAO WANG[4], AND KAITAI LIANG[3]

[1]Beijing Key Laboratory of Trusted Computing, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China
[2]Tianjin Key Laboratory of Advanced Networking, College of Intelligence and Computing, Tianjin University, Tianjin 300072, China
[3]Surrey Centre of Cyber Security, University of Surrey, Guildford GU2 7XH, U.K.
[4]Department of Computer Science, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway

Corresponding authors: Guangquan Xu (losin@tju.edu.cn) and Naixue Xiong (xiongnaixue@gmail.com)

This work was supported in part by the by National Natural Science Foundation of China under Grant 61501007, Grant 61572355, and Grant U1736115, in part by the Beijing Postdoctoral Research Foundation under Grant 2017-22-030, and in part by the CCF-Venustech Open Research Fund under Grant CCF-VenustechRP2017008.

ABSTRACT In the Internet of Things, based on the collaboration of sensing nodes, sensing data are collected and transmitted. The collaboration of sensing nodes also plays an important role in the safeguard of the Internet of Things. Owing to the limited ability of the single sensing node, the threshold authentication based on the collaboration of sensing nodes can improve the trust of security authentication of sensing nodes. The current threshold authentication schemes may require high-computational complexity, and more importantly, most of them are instantiated by membership authentication. It's challenging to apply the current state of the arts to the case where sensing nodes with various weights join together to fulfill a relatively lightweight authentication. In this paper, we first design a communication key distribution scheme for sensing networks based on a symmetric operator. Using the permutation function, the scheme is able to generate characteristic sequences to improve the efficiency of key distribution in sensing networks. In addition, we propose a threshold authentication scheme based on weights, in which the higher weight represents the more important role in authentication. Our authentication scheme only requires lightweight operations, so that, it is extremely friendly to the IoT nodes with restricted computation power. The security analysis and the case verification demonstrate that our novel authentication protects IoT nodes without yielding significantly computational burden to the nodes.

INDEX TERMS Internet of Things, symmetric operator, weighted threshold authentication, threshold equation.

I. INTRODUCTION

Internet of Things (IoT) is an advanced framework to connect a considerable amount of objects (with WiFi or 4G communication functionalities), in practice, to the Internet, so as to fill the technical gap between the Internet and the terminal [1]–[10]. Compared with the traditional Internet, IoT may assemble many heterogeneous sensor networks that make the network security more sophisticated and further difficult to be analyzed. Being one of the secure underpinnings of IoT, security authentication has become a research hotspot in recent years.

Security authentication is one of the core mechanisms used in the perceptual layer. Many cryptographic tools are able to be employed to achieve authentication protocols, e.g., symmetric-key authentication [11], public-key authentication [12]–[14], key-shared authentication [15], dynamic user authentication [16], [17] and many others [18]–[24]. Public-key authentication (PKA) mainly includes a hybrid-key management protocol based on the elliptic curve cryptography (ECC) and the symmetric-key cryptography [12], the RSA-based TinyPK authentication scheme [13] and the distributed public-key authentication cooperative scheme [14]. A PKA protocol could be deployed in the perceptual layer to guarantee secure authentication among sensor nodes. Due to a large number of exponential operations incurred by asymmetric encryption technique, the overhead

The associate editor coordinating the review of this manuscript and approving it for publication was Chunsheng Zhu.

of IOT is extremely increased along with the number of nodes.

Symmetric-key authentication (SKA) consists of several secure ingredients. For example, security framework protocol SPINS [11] consists of network security encryption protocol SNEP, time-based high-tolerant loss packet flow-based protocol μTESLA, multi-level μTESLA authentication protocol and one-way key chain authentication scheme. Different from heavy computational complexity required by PKA, SKA only requires lower computation (i.e. low energy consumption), which is a better candidate for sensor nodes. However, since all symmetric keys are shared by sensor nodes and sink node, if sink node has the single node invalidation, all symmetric keys may leak

In [15], a secret-shared distributed authentication protocol is proposed to effectively reduce the calculation and energy consumption in the encryption and decryption stages of perceived nodes. However, it is necessary to design a mechanism to coordinate information transmission and reception. In [16] and [17], a dynamic user authentication protocol is proposed, which uses one-way hash function and logical XOR operation instead of key encryption, therefore improving the computation efficiency of nodes. However, as sink node is required to play the role of information center during authentication, it will bring a heavy burden for sink node. In addition, [25]–[29] focus on achieving the balance between the efficiency and the communication cost, but less progress is obtained.

Since one needs to always balance both limited computing tasks and computational power of the IoT nodes, the security and the efficiency are challenging to be achieved at the same time in the aforementioned authentications. In this paper, we first design a key distribution scheme for sensing networks based on symmetric operator. Using the permutation function, the scheme is able to generate characteristic sequences to improve the efficiency of key distribution in sensing networks. In addition, we propose a threshold authentication scheme based on weights, which only requires limited operation complexity, being very friendly to the IoT nodes. The outline of the paper is described as follows.

The rest of this paper is organized as follows: Section II introduces the related work. In section III, we propose a communication key assignment scheme based on symmetric operator. Section IV constructs the threshold authentication scheme based on weights. In Section V, we improve the threshold authentication scheme based on weights and propose two improved threshold authentication schemes based on weights. Finally, we give the conclusion and future work in Section VI.

II. RELATED WORK

Traditionally, security authentications in IOT focus on perceptual level security, protecting data on transmission among sensor nodes [30]. The simplest approach to perceptual level security consists of using a network-wide encryption key, such that ZigBee authentication [31]. ZigBee also provides support for cluster and individual link keys. MiniSec [32] is another well-known security authentication for IOT that provides data confidentiality, authentication and replay protection. As with ZigBee, the packet overhead introduced by MiniSec is in the order of a few bytes. The widespread TinySec perceptual layer security mechanism is no longer considered secure [32].

Most security authentications of IOT do not consider a protocol for key distribution to sensor nodes. Keys are either loaded onto the nodes before setup or a separate key establishment protocol is used [33]–[35]. Public key cryptography (PKC) is used in traditional computing to facilitate secure key establishment. However, PKA, in particular the widespread RSA algorithm, involves too resource consuming for constrained devices. Some security authentications, such as Sizzle [36], advocate the use of the more resource efficient Elliptic Curve Cryptography (ECC) public key cryptosystem. Other research efforts, such as the secFleck [37] mote, provide support for faster RSA operations through hardware.

Approaches without PKC often rely on the pre-distribution of connection keys. Random key pre-distribution schemes, such as the q-composite scheme [38], establish connections with a node's neighbors with a certain probability. Intuitively, pre-distributed key schemes such as this require a large amount of keys to be loaded onto the nodes before deployment. Depending on the method used, this approach is scaling in $O(n^2)$ or $O(n)$, where n is the number of nodes in the network. The Peer Intermediaries for Key Establishment authentication achieves sub linear scaling in $O(\sqrt{n})$ by relying on the other nodes as trusted intermediaries. While PIKE provides higher memory efficiency than random schemes, it still leaks additional key information when motes are captured.

In order to reduce the burden of authentication in sensor networks, end-to-end security authentications have been widespread studied. Classical end-to-end security authentications are Sizzle [36], SSNAIL [39] and DTLS [40]. As outlined in the introduction, such a authentication protects the message payload from the data source until it reaches its target. Because end-to-end authentications are usually implemented in the network or application layer, forwarding nodes do not need to perform any additional cryptographic operations since the routing information is transmitted in the clear. On the flip side, this means end-to-end security protocols do not provide the same level of protection of a network's availability as a perceptual layer protocol could.

Security authentication for single sensor node cannot guarantee the trust of this node. Because IOT is vulnerable to network attacks and sensor nodes are vulnerable to be attacked or be captured. So, threshold authentication is involved to guarantee the trust of sensor networks. Only sensor nodes with a number exceed the threshold can achieve the authentication. Authentication with sufficiently number of sensor nodes has more trust compared with authentication with single node.

The threshold cryptography is a technique that the encryption key or signature key is dispersed into the entity or component by secret sharing technology. Only the combination of

entities or components with no less than a threshold can perform an encryption or signature function [41]–[44]. A typical threshold signature is (k, n) threshold signature [45]–[48], where only a combination of larger than or equal to k entities can be signed and otherwise signing leads to a failure. Based on this technique, in [49], a threshold signature with privilege set proposed, where only a sufficient number of privileged set members participating in the signature can form a valid signature, while any signature that is involved in a privileged set less than the number of the threshold, even with the participation of more ordinary members, cannot be valid. In [50], a signature scheme based on weighted threshold is proposed. Each member plays different roles in the signature scheme dependent on their weights. However, because of using the Chinese Remainder Theorem in the signature construction, the scheme requires large number of multiplications and exponentiation operations in signature, yielding significantly computational burden to IoT nodes.

Binary symmetric polynomial key distribution is widely used for secure communication in perceptual networks. The mechanism works as follows [51]. Select the binary symmetric polynomial $f(x, y) = \sum_{i,j=0}^{t} a_{ij}x^i y^j$ on the prime field, where $f(x, y) = f(y, x)$. We suppose ID_1, ID_2 are the identity of two perceived nodes.

If $f_2(x) = f(x, ID_2)$, $f_1(x) = f(x, ID_1)$, assign the values to ID_1, ID_2, respectively. We have the following result, $f_1(ID_2) = f(ID_1, ID_2) = f_1(ID_2, ID_1) = f_2(ID_1)$. Therefore, the ID_1, ID_2 shared secret key is $f(ID_1, ID_2) = f_1(ID_2, ID_1)$.

Since the design is based on the prime number domain, the computational complexity of two variables symmetric polynomial increases rapidly with the increase of threshold, which limits its practical use in sensing networks.

III. COMMUNICATION KEY ASSIGNMENT BASED ON SYMMETRIC OPERATOR

A. DEFINITION OF SYMMETRIC OPERATOR

Definition 1: Operator $H : R^m - > R$ is said as symmetric operator If only: $\forall 1 \le i, j \le m$, there is:

$$H(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_m) = H(x_1, \ldots, x_j, \ldots, x_i, \ldots, x_m) \quad (1)$$

Assume that there are L sensing nodes in the sensing networks, which donated as: $N_1, N_2, \ldots\ldots, N_L$, each of them has an $ID_i, i \in [1, L]$. The key distributed management center KDS chooses the symmetric operator as the key generation operator of the whole sensing networks and delivers the sub symmetric operator for each sensing node: $H_i = H(\ldots\ldots, ID_i, \ldots\ldots)$. Then any m nodes in the sensing networks share the key

$$k = H_1(ID_1, ID_2, ID_3, \ldots\ldots\ldots\ldots, ID_m). \quad (2)$$

The binary symmetric polynomial also satisfies the properties of the symmetric operator. In order to improve the computation efficiency, we illustrate a symmetric operator based on $GF(2^n)$. The basic form of the element in $GF(2^n)$ domain is described as follows:

$$f(x) = \sum_{i=1}^{t} a_{i-1}x^{i-1}, \quad a_i \in GF(2^n).$$

$$g(x) = \sum_{i=1}^{t} b_{i-1}x^{i-1}, \quad b_i \in GF(2^n).$$

The addition and multiplication formula in $GF(2^n)$ domain is defined as:

$$f(x) + g(x) = \sum_{i=1}^{t} (a_{i-1} \oplus b_{i-1})x^{i-1}$$

$$f(x) \times g(x) = \sum_{i=1}^{t} a_{i-1}x^{i-1}g(x) \bmod P(x),$$

where $P(x)$ is the prime polynomial of $GF(2^n)$.

B. COMMUNICATION KEY ASSIGNMENT

To improve the efficiency of the identity authentication between sensing nodes, we propose a hierarchical key assignment scheme based on symmetric operator, which includes the hierarchical authentication procedure from the cluster head node to ordinary sensing node as well as the same procedure from sink node to cluster head node. Suppose there are n clusters in the sensing networks, and the cluster i has n_i sensing nodes.

1) The key distributed manager KDS selects the non-degenerate matrix with $(n + 1)$ order:

$$A_i = \begin{pmatrix} a_{i11} & \ldots & a_{i1m_i} \\ \ldots & \ldots & \ldots \\ a_{im_i1} & \ldots & a_{im_im_i} \end{pmatrix}, \quad (3)$$

where $\forall 0 \le i \le n, 1 \le j, k \le m_i, a_{i,j,k} = a_{i,k,j}$.

We let $A = \{A_0, \ldots\ldots, A_n\}$ be the key matrix of the sensing networks, and the key matrix A can be updated. The element of the key matrix A is defined in $GF(2^n)$.

Assume X, Y are arbitrary m_i dimension vectors, then we define $H_i(X, Y) = X^T A_i Y$. It is obvious that $H_i(X, Y) = X^T A_i Y = Y^T A_i X = H_i(Y, X)$ therefore H_i is symmetric operator.

2) Suppose that the sink node and all cluster head nodes are identified as follows: $(ID_0, ID_1, \ldots, ID_n)$, where ID_0 is the sink node and others are cluster head nodes. Assume that the nodes in the cluster network are identified as $(ID_i, ID_{i,1}, \ldots, ID_{i,n_i})$, where ID_i is the cluster head nodes identifier, and the rest of them are sensing nodes.

Then the characteristic sequence of each node (sink node, cluster head node and ordinary node) can be defined as follows:

$$Tr(ID) = ((T\_ID)_1, \ldots, (T\_ID)_{m_i})^T, \quad (4)$$

where $Tr : R \to R^{m_i}$ is permutation function. Let *Table* be a secure single-byte permutation table, *Circle* is a circulation towards the right and $C \in GF(2^n)$ is a constant.

TABLE 1. Computational comparison between binary symmetric polynomial scheme [51] and ours.

	Our Scheme	[51]
The computation of the private key generated in all nodes authentication	$n(m^2+c)G$	$n(m^2+m)P$
The computation of the shared private key generated by any two nodes	$(m+c)G$	$2mP$

Algorithm 1

Set $T\_ID_1 = ID \oplus C$
For $(j = 0; j < m_i - 1; j++)$
{
$x = Circle(T\_ID_j, step)$;
$y = x \oplus C$;
$T\_ID_{j+1} = Table(x)$;
}

The above characteristic sequence has higher efficiency than the characteristic sequence $(ID^1, \ldots, ID^{m_i})^T$ of the binary symmetric polynomial key distribution scheme.

3) The key distribution manager KDS generates the authentication key as follows

The authentication key of node ID_i between sink node and cluster head node is

$$K_i = H_0(Tr(ID_i), .) = A_0 Tr(ID_i). \tag{5}$$

The authentication secret key of cluster head node $ID_i (i \neq 0)$ in the cluster network i is

$$\hat{K}_i = H_i(Tr(ID_i), .) = A_i Tr(ID_i). \tag{6}$$

The authentication secret key of sensing node $ID_{i,j}$ in the cluster network i is

$$\hat{K}_{i,j} = H_i(Tr(ID_{i,j}), .) = A_i Tr(ID_{i,j}). \tag{7}$$

The key distribution manager KDS encrypts the authentication key with the permanent key of member $ID_i(ID_{i,j})$ and sends it to $ID_i(ID_{i,j})$. $ID_i(ID_{i,j})$ decrypt the message and get authentication private key. By using this method, any two members in the group will generate their private key shared by each other, which donated as:

$$\begin{aligned} k_{i,j} &= Tr(ID_j)^T K_i = Tr(ID_j)^T A Tr(ID_i) \\ &= Tr(ID_i)^T A Tr(ID_j) = Tr(ID_i)^T K_j = k_{j,i}. \end{aligned} \tag{8}$$

Since the order of key matrix A_i in the perceptual network is m_i, the proposed scheme is a key management scheme with threshold property. In order to ensure the unconditional security of the communication between the sink node and the cluster head node, we can suppose that

$$m_0 \geq n + 2. \tag{9}$$

At this point, even if the sink node as well as all other cluster nodes is destructed, the key matrix A_0 cannot be completely stolen.

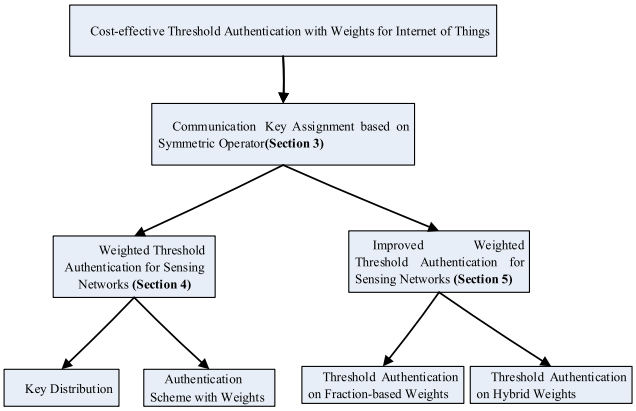


FIGURE 1. The architecture of this paper.

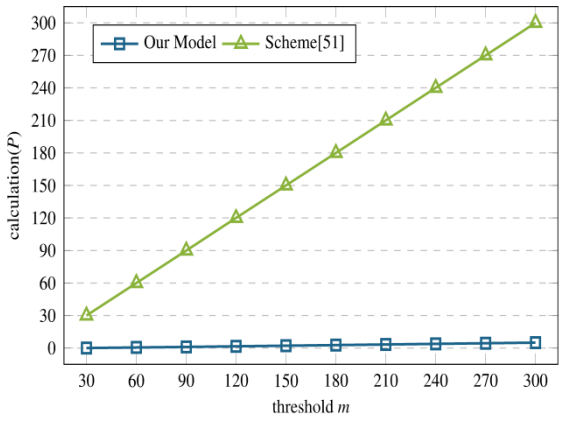


FIGURE 2. Computation comparison w.r.t. generation of characteristic sequences.

C. COMPUTATION EFFICIENCY COMPARISON

Table 1 illustrates the comparison between the efficiency of key allocation scheme in this paper and the existing binary symmetric polynomial key allocation scheme if the number of the nodes is n and the threshold is m:

In table 1, $c \ll 1$ is a constant, and G, P represent a modular multiplication on $GF(2^n)$ and a prime field respectively. We know that $G < P$. It can be predicted from Table 1 that the efficiency of this scheme is higher than that of the binary symmetric polynomial key allocation scheme, whether the KDS generates the authentication private key of all nodes or the shared private key from one node to other sensing nodes. Fig.2 illustrates the computations with respect to value

m, described by two characteristic sequences generated by the binary symmetric polynomial and the proposed scheme. As we can see from Fig.2, the computation complexity (The abscissa unit is P) of the characteristic sequence obtained in this paper is much lower than that of the binary symmetric polynomial key allocation scheme. Therefore, the scheme has better computation efficiency.

Since the $Tr(ID) = ((T\_ID)_1, \ldots, (T\_ID)_{m_i})^T$ of different ID are scattered randomly, the probability of this characteristic sequence linear associated with any m sensing nodes is $1/2^k$, which k is the number of bits of ID.

Then the probability of linear correlation of any m nodes in M sensing nodes set is:

$$\Pr = \frac{C_M^m}{2^k}.$$

Considering the practical situation, we set $k = 128$, $m = 15, M = 200$, then we can infer that $\Pr \leq 2^{-54}$, this value could certainly be ignored.

IV. WEIGHTED THRESHOLD AUTHENTICATION FOR SENSING NETWORKS

In this section, we propose threshold authentication scheme based on weights. The higher of weight stands for the more important role in authentication. Moreover, the proposed scheme is based on the symmetric operator key assignment scheme proposed in the previous section. It only uses the multiplication on the finite polynomial domain to match the computation power of each sensing node and reduce the burden of these sensing nodes.

A. BASIC DEFINITION

According to the discussion in section 3, we assume the sensing node set $N_1, \ldots\ldots, N_m$ with identifiers $ID_1, \ldots\ldots, ID_m$. The weight of each sensing node N_i is derived from the $V = \{V_1, \ldots\ldots, V_t\}$ (V_i is positive an integer). Each node has only certain level of weights (which can be dynamically adjusted), and multiple nodes may have the same level of weights. We suppose that the sensing nodes $N_1, \ldots\ldots, N_m$ have c_i sensing nodes with confidence V_i, the identities of these nodes are $ID^i_1, \ldots\ldots, ID^i_{c_i}$. Define $TV_i = \{ID^i_1, \ldots\ldots, ID^i_{c_i}\}$, where TV_i is the set of all sensing nodes with weight V_i in $N_1, \ldots\ldots, N_m$.

Definition 2: Sensing nodes $N_1, \ldots\ldots, N_{k_l}$ complete the threshold signature together based on weight, if the weight of the participating nodes is greater or equal to K, which is expressed as:

$$x_1V_1 + \ldots\ldots + x_tV_t \geq k, \tag{10}$$

where k is a positive integer which represents the threshold value.

The proposed scheme becomes a typical (k, n) threshold signature only if $V_1 = \ldots\ldots = V_t = V_0$. In another word, only the union with k/V_0 or more sensing nodes can launch the signature process, which in turn means any union contained less than k/V_0 nodes cannot complete the signature at all.

If $V_1 \gg V_2 + \ldots\ldots + V_t$, the scheme in this paper will become a threshold signature with privilege set V_1. That means the sensing node union which contains the threshold value of the privilege set V_1 or more of that can launch the signature process. In another word, those unions with less than the threshold value of the privileged set nodes cannot complete the signature procedure no matter how many other nodes participate in.

Suppose that group G is an order q multiplication generation group with generator g, q is a large prime number which donated as $q = 2^n - 1$, Z_q is a domain. $H : \{0, 1\}^* \to Z_q$ is a security-encoded hash function. Obviously $GF(2^n)\backslash 0$ is a $2^n - 1$-order multiplication cyclic group.

B. THRESHOLD AUTHENTICATION WITH WEIGHTS

Consider the weighted authentication scheme of sensing networks with threshold equation.

$x_1V_1 + \ldots\ldots + x_tV_t \geq k$, $V_1, \ldots\ldots, V_t, k$ are positive integer, and $V_1, \ldots\ldots, V$ are relatively prime. For further discussion, the threshold equation discussed above is called an integer weight threshold equation.

1) KEY DISTRIBUTION

The key distribution manager KDS selects the polynomial $f(x)$ defined on Z_q

$$f(x) = sk + \sum_{i=1}^{k} a_{t-1}x^{t-1},$$

where $f(0) = sk$ is the shared key. Calculate and distribute the public key $pk = g^{sk}$.

Let $c = \sum_{i=1}^{t} \min\{\lfloor c_iV_i \rfloor, k\}$.

The key distribution manager KDS selects primary id $id_1, \ldots\ldots, id_c$ (these id are different positive integers), and establish the key fragments as follows: $S_1, \ldots\ldots, S_c$.

1) $S_i = f(id_i)$ $i = 1, 2, \ldots\ldots\ldots, c$.

2) m_i is a minimum positive integer that satisfies $m_iV_i \geq k$

$$p_i = \sum_{j=1}^{i-1} m_jV_j, \quad 1 \leq i \leq t,$$
$$k_i = \min\{c_iV_i, m_iV_i\}.$$

The key distribution manager KDS selects key fragments from $S_{p_i+1}, \ldots\ldots, S_{p_i+k_i}$, and deliver them to the sensing nodes in the TV_i. Each node obtains V_i key fragments sequentially. That means KDS assigns the key $S_{p_i+V_i(j-1)+1}, \ldots\ldots, S_{p_i+V_ij}$ to the node ID^i_j For each node $N_i(i = 1, \ldots, m)$, KDS chooses single signature key sk_i, calculate $r_i = g^{sk_i}$, adds (ID_i, r_i) in the signature public information list and sends sk_i to N_i by secure tunnel. Different from the shared key sk, single-signed key sk_i needs to be regenerated in each signature process. In order to reduce the burden of sensing nodes beyond normal interaction, KDS uses mass generation or mass-produce techniques.

2) PROTOCOL DESIGN

Suppose that the sensing nodes $N_1, \ldots\ldots, N_{k_l}$ participate in the signature process, where the number of these sensing nodes in TV_i set is c_i'. H is the security-encoded hash function. Then we assume that λ_i is the Lagrangian multiplier corresponding to the primary identification $N_1, \ldots\ldots, N_{k_l}$ which we have mentioned before. A whole threshold authentication scheme includes the following steps.

1) We suppose that the nodes in TV_i set which also participate in the signature process are $ID_1^i, \ldots\ldots, ID_{c_i'}^i$.

where sk_j^i is the private key in single-signed process of ID_j^i. In summary, ID_j^i calculates $s_j^i = \sum_{z=1}^{j} s_{p_i+V_i(j-1)+z}$ in each node ID_j^i. After that, ID_j^i sends the signature result to Sink node.

2) Sink node collects signature fragment s_i^j of all nodes, combines them with m and sends this final signature to the challenge party.
3) The challenger calculates the sum of all signature fragments as $s = \sum s_j^i$ and verifies this process as follows:

$$r = \prod_{i=1}^{k_l} r_i,$$

$$g^s r = g^{\sum s_j^i} r = g^{\sum_{i=1}^{k} \lambda_i f(id_i) H(m)} g^{-\sum_{i=1}^{k_l} sk_i} g^{\sum_{i=1}^{k_l} sk_i} = g^{f(0)H(m)} = pk^{H(m)}.$$

That is $g^s r = pk^{H(m)}$.

3) FORMAL SECURITY ANALYSIS

It is obvious that this scheme can cover all possible integer solutions of the indeterminate equation $x_1 V_1 + \ldots\ldots + x_t V_t \geq k$. It is pretty clear that any union of the nodes whose sum of weights is less than the threshold value should not complete the signature procedure. In another word, the unions whose sum of weights is equal to the threshold or more than that could complete the signature procedure. Thus, the scheme can resist the collusion attack from any union whose sum of weights is less than the threshold value. Based on the threshold and discrete logarithm problems, the proof progress of this scheme is illustrated in detail as follows:

The security analysis is based on the discrete logarithm problem:

Discrete Logarithm Problem: For large integer group, after given the random number $z \in G$, it is difficult to find $r(r > 1)$ to satisfy the equation $g^r = z$.

The security proof procedure of the proposed scheme can be illustrated by the following theorems:

Theorem 1: We assume that there exists an adaptive chosen message and identity attacker F can break this Each node ID_j^i computes separately the following formulas.

$$\begin{aligned} s_{p_i+V_i(j-1)+1} &= \lambda_{p_i+V_i(j-1)+1} S_{p_i+V_i(j-1)+1} H(m), \\ s_{p_i+V_i(j-1)+2} &= \lambda_{p_i+V_i(j-1)+2} S_{p_i+V_i(j-1)+2} H(m), \\ &\ldots\ldots\ldots\ldots\ldots\ldots \\ s_{p_i+V_ij} &= \lambda_{p_i+V_ij} S_{p_i+V_ij} H(m) - sk_j^i \end{aligned}$$

scheme in PPT time with probability ε which cannot be ignored. Then there exists an algorithm C who can solve the problem of discrete logarithm in the PPT time with a probability ε that cannot be ignored. $O(\varepsilon)$ represents ε is not less than a constant which is related to the random oracle function q_{H_1}, q_{H_2}, q_L, with no relation with the safety parameter n.

Proof: We assume that C is a challenger, the goal of C is eventually output a solution to the discrete logarithm problem through calling F.

(1) C runs the setup algorithm. Challenger C maintains the signature public keys sequence $pk_1, \ldots, pk_t$, and constructs two oracles H_1, H_2 (The construction of H_1, H_2 is shown below). C sends the data $\{pk_1, \ldots, pk_t, H_1, H_2\}$ to the attacker F as a public parameter.
(2) H_1 inquiring process. C maintains a list H_1^L which contains the arrays $\{m_i, h_i\}$ as well as randomly prepares q_{H_1} responses donated as $h_1, \ldots, h_{q_{H_1}}$. When F accesses the H_1 value of m_i, C recovers the $\{m_i, h_i\}$ item from the list H_1^L and sends h_i to F.
(3) H_2 inquiring process. C maintains a list H_2^L containing the arrays $\{m_i, \widehat{h}_i = \prod_{j=1}^{k_l} r_{i,j}\}$ as well as randomly prepares q_{H_2} responses $\widehat{h}_1, \ldots, \widehat{h}_{q_{H_2}}$. When F accesses H_2 value of the m_i, C recovers the $\{m_i, \widehat{h}_i\}$ item from the list H_2^L and sends $\widehat{h}_i$ to F.

Signature inquiring process. C maintains a list L^L containing q_L arrays $\{m_i, s_i\}$ and F inquiry a signature to s_j, and C checks whether the s_j is in the list s_j, then recovers the s_j item and sends s_j to F.

The attacker F interacts with C, while C outputs its responses according to the policy above.

When F stops inquiring the result, F outputs a signature s_j related with m_i (The signature of m_i has never been asked before), which also satisfies $Ver(m_j, s_j) = 1$. C recovers item $\{m_j, h_j\}$ from the list H_1^L, and recovers other items $\{m_j, \widehat{h}_j\}$ from the list H_2^L. If $z = pk^{H(m_j)}/\widehat{h}_j$, then we could predict that $g^s = z$. That means we finally solve the discrete logarithm problem.

If $z = pk^{H(m_j)}/\widehat{h}_j$ equals to a certain m_j that has been asked before, we can obtain the probability of that is $\frac{q_L}{2^n}$ according to drawer principle. Therefore, the probability of successfully solve the discrete logarithm problem by C remains the same, donated as $\varepsilon' = O(\varepsilon) + \frac{q_L}{2^n} = O(\varepsilon)$.

4) CASE ANALYSIS

We suppose that $V = \{2, 3\}$ is the weight set of sensing nodes $ID = \{1, \ldots\ldots, 8\}$. Weighted threshold authentication is defined as follows:

$$2x_1 + 3x_2 \geq 7.$$

We select the following polynomial:

$$f(x) = 11 + x^6 \bmod 13,$$

where $f(0) = 11$ is the shared key. Let the primary identify $id_1, \ldots\ldots, id_{20} = 1, \ldots\ldots, 20$, and calculate:

$$S_1 = f(id_1) = 12, S_2 = f(id_{20}) = 10,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$S_{19} = f(id_{19}) = 10, S_{20} = f(id_{20}) = 10.$$

Denote $TV_1 = \{1, \ldots\ldots, 4\}$, $TV_2 = \{5, \ldots\ldots, 8\}$. Then we deliver $S_1, \ldots\ldots, S_8$ to the sensing nodes of TV_1, each node obtains two keys. And deliver $S_9, \ldots\ldots, S_{20}$ to the sensing nodes of TV_2, each node get three keys.

The core of the threshold signature is the indirect recovery of the threshold key in the operation procedure. The key recovery process in this scenario is described as follows: we assume that (2, 1) is the solution of the equation, in another words, there are two nodes and a node involved in the key recovery from TV_1 and TV_2, respectively. Above all there are total S_i of number 7 involved in key recovery process. Because the power number of $f(x)$ is 6, key $f(0) = 11$ is restored after using the Lagrange theorem.

5) VALIDITY

In Table 2, we compare the efficiency of the proposed scheme with the existing schemes, with m nodes participated in the signature and the average weight V_0.

TABLE 2. Comparison our scheme with [50].

	Key distribution phase	Signature phase	Verification phase
Ours	$O(mE)$	$O(mV_0(k+1))$	E
[50]	$O(mE)$	$O(mE)$	$O(mE)$

In the three phases mentioned before, only signature phase is completed by sensing nodes. E represents exponential operation of large numbers, k represents the threshold. Set $id_1, id_2, \ldots\ldots = 1, 2, \ldots\ldots$, that is, the primary identify is integer series beginning with 1, and $V_0(k+1) \ll E$.

We can see from table 2, the computation complexity of the proposed scheme, especially the signature phase performed by the sensing nodes, is much lower than that of the scheme [50], and it can match the computational power of the sensing nodes.

As we can see from Fig.3, the computation complexity (The abscissa unit is E) of the signature phase is much lower than that of scheme [50]. For example, when the average weight of sensor nodes is 50, the average calculations of scheme [50] is about 50E, however the average calculations of the proposed method is about 2.5E, which is much lower than scheme [50]. Therefore, the scheme has better computation efficiency.

V. IMPROVED WEIGHTED THRESHOLD AUTHENTICATION FOR SENSING NETWORKS

In the previous section, we propose a weighted authentication scheme with integer weight threshold. It can solve the problem of joint verification of nodes with different weighted levels, and it is more efficient compared with those existed signature schemes. However, we can see from Table 1 that the burden of sensing nodes was rising rapidly with the increasing range of weights $V = \{V_1, \ldots\ldots, V_t\}$. For instance: considering the following threshold equation:

$$77x_1 + 91x_2 + 143x_3 \geq 2002.$$

Now, the key management center exploits those polynomials with 2002 power to assign the sub keys to each sensing node. That means the calculation of multiplication have to allocate on average, which could create burden for each sensing node.

Transform the formula above through divided by about 1001 inequality, and then we get:

$$\frac{x_1}{13} + \frac{x_2}{11} + \frac{x_3}{7} \geq 2.$$

The related parameters of this threshold equation are much smaller than those of the original equation. Therefore, we focus on a more efficient threshold proof scheme for the above equations in this section.

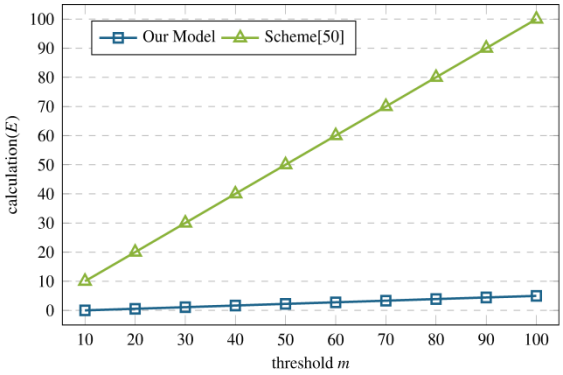


FIGURE 3. Computation comparison w.r.t. the generation of characteristic sequence.

A. THRESHOLD AUTHENTICATION BASED ON FRACTION WEIGHTS

Considering fraction-based weights threshold equation as follows:

$$\frac{x_1}{M_1} + \ldots\ldots + \frac{x_t}{M_t} \geq k \tag{11}$$

where $M_1, \ldots\ldots, M_t, k$ are positive integers and $M_1, \ldots\ldots, M_t$ are inter-prime. The threshold equation with integer weights corresponding to the fraction-based weights threshold equation is expressed as:

$$M_2, \ldots, M_r x_1 + \ldots\ldots + M_1, \ldots, M_{t-1} x_t \geq kM_1, \ldots, M_t$$

According to the description in 4.1, TV_i is subset of sensing nodes set $N_1, \ldots\ldots, N_m$, whose weights are

$$M_1 \cdots M_{i-1} \cdot M_{i+1} \cdots M_t.$$

In order to describe more clearly in the subsequent sections, TV_i is defined as a set of sensing nodes corresponding to x_i, with the size of this set is c_i.

1) KEY DISTRIBUTION

The key distribution manager KDS exploits the polynomial $f(x)$ defined on Z_q which donated as:

$$f(x) = sk + \sum_{i=1}^{k} a_{t-1} x^{t-1}.$$

We assume that $f(0) = sk$ is the shared key. KDS calculates and published the public key $pk = g^{sk}$.

Suppose that:

$$c = \sum_{i=1}^{t} \min\{\lfloor c_i/M_i \rfloor, k\}.$$

The key distribution managers exploit primary identity $id_1, \ldots\ldots, id_c$ to generate the key fragments $S_1, \ldots\ldots, S_c$ as follows:

1) $S_i = f(id_i)$ $1 \le i \le c$,

2) We suppose that:

$$p_i = \sum_{j=1}^{i-1} \min\left\{\left\lfloor c_j/M_j \right\rfloor, k\right\} \quad 1 \le i \le t$$
$$k_i = \min\left\{\left\lfloor c_{i/M_i} \right\rfloor, .k\right\}.$$

3) The key distribution manager exploits the key fragments $S_{p_i+1}, \ldots\ldots, S_{p_i+k_i}$ to generate the key again, then deliver the fragments to the nodes in the TV_i in turn. The polynomial selected by the key distribution manager KDS is expressed as:

$$\begin{aligned} f_{p_i+1}(x) &= S_{p_i+1} + a_{p_i+1,1}x \\ &\quad + \ldots\ldots + a_{p_i+1,M_i-q_i-1}x^{M_i-q_i-1}, \\ f_{p_i+2}(x) &= S_{p_i+2} + a_{p_i+2,1}x \\ &\quad + \ldots\ldots + a_{p_i+2,2M_i-q_i-1}x^{2M_i-q_i-1}, \\ f_{p_i+k_i}(x) &= S_{p_i+k_i} + a_{p_i+k_i,1}x \\ &\quad + \ldots\ldots + a_{p_i+k_i,k_iM_i-q_i-1}x^{k_iM_i-q_i-1}. \end{aligned}$$

For each node ID_j^i in TV_i set, we calculate that

$$\begin{aligned} S_{p_i+1}^j &= f_{p_i+1}(ID_j^i), \\ S_{p_i+2}^j &= f_{p_i+2}(ID_j^i), \\ S_{p_i+k_i}^j &= f_{p_i+k_i}(ID_j^i). \end{aligned}$$

Deliver the $S_{p_i+1}^j, \ldots\ldots, S_{p_i+k_i}^j$ to the node ID_j^i.

For each node $N_i (i = 1, \ldots, m)$, we use KDS to select a single-signed key sk_i and calculate $r_i = g^{sk_i}$. Then we add (ID_i, r_i) item to the public signature list and send sk_i to N_i through the secure channel. Different from the generation of shared key, a single-signed key sk is generated in each signature generating process. In order to reduce the burden in each sensing node beyond normal interaction, KDS uses mass generation or mass-produce techniques.

2) PROTOCOL DESIGN

Suppose that the sensing nodes $N_1, \ldots\ldots, N_{k_l}$ participate in the signature process, where the number of these sensing nodes in TV_i set is c_i'. H is the security-encoded hash function. Then we assume that λ_i is the Lagrangian multiplier corresponding to the primary identification $N_1, \ldots\ldots, N_{k_l}$ which we have mentioned before. Our construction is described as follows.

1) Select $f_{p_i+1} \cdot \cdot, f_{p_i+c_i'/M_i}$ as the polynomials that participate in signature procedure.

We suppose that the sensing nodes participated in signature process in TV_i are $ID_1^i, \ldots\ldots, ID_{c_i'}^i$ and then we compute as follows:

The Lagrange multipliers corresponding to the first M_i nodes are $\lambda_{p_i+1,j}$,

The Lagrange multipliers corresponding to the first $2M_i$ nodes are $\lambda_{p_i+2,j}$,

The Lagrange multipliers of all nodes are $\lambda_{p_i+c_i'/M_i,j}$.

For each node ID_j^i, we calculate that:

$$\begin{aligned} s_{p_i+\lfloor j/M_i \rfloor+1} &= \lambda_{p_i+\lfloor j/M_i \rfloor+1}\lambda_{p_i+\lfloor j/M_i \rfloor+1,j} f_{p_i+\lfloor j/M_i \rfloor+1} \\ &\quad \times (ID_j^i)H(m), \\ s_{p_i+\lfloor j/M_i \rfloor+2} &= \lambda_{p_i+\lfloor j/M_i \rfloor+2}\lambda_{p_i+\lfloor j/M_i \rfloor+2,j} f_{p_i+\lfloor j/M_i \rfloor+2} \\ &\quad \times (ID_j^i)H(m), \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ s_{p_i+k_i} &= \lambda_{p_i+k_i}\lambda_{p_i+k_i,j} f_{p_i+k_i}(ID_j^i)H(m) - sk_j^i. \end{aligned}$$

sk_j^i is the single-signed key of ID_j^i, and the node ID_j^i calculates

$$s_j^i = \sum_{z=p_i+\lfloor j/M_i \rfloor+1}^{p_i+k_i} s_z.$$

The node ID_j^i sends the signature result s_j^i to Sink node.

2) Sink node collects all signature fragments $V = \{3, 5\}$ of all nodes, combines them with $V = \{3, 5\}$ and sends this final signature to the challenger party.

3) The challenger calculates the sum of all signature fragments, donated as $s = \sum s_j^i$ and verifies this process as $g^S r$, shown at the bottom of this page. This procedure can also be expressed as: $g^s r = pk^{H(m)}$.

$$\begin{aligned} g^S r &= g^{\sum\sum_{V_i}\sum \lambda_{p_i+\lfloor j/M_i \rfloor+z}{}^{lambda} p_i+\lfloor j/M_i \rfloor+z, j f\, p_i+\lfloor j/M_i \rfloor+z}{}^{(ID_j^i)H(m)} g^{-\sum_{i=1}^{k_l} sk_i} g^{-\sum_{i=1}^{k_l} sk_i} = g^{-\sum_{i=1}^{k_l} \lambda_i f(id_i)H(m)} \\ &= g^{f(0)H(m)} = pk^{H(m)} \end{aligned}$$

3) FORMAL SECURITY ANALYSIS

It is obvious that any $x_1, \ldots, \ldots, x_t$ satisfied the equation

$$\left\lfloor \frac{x_1}{M_1} \right\rfloor + \ldots\ldots + \left\lfloor \frac{x_t}{M_t} \right\rfloor \geq k_c$$

can obtain k values of $f(x)$, so the signature process can be achieved. According to the features of key distribution, signature combinations that do not satisfy the threshold equation will not have enough key fragments. Moreover, they cannot be encrypted, signed and execute other steps. Therefore, this program can resist any collusion attack that is less than the threshold value.

4) CASE ANALYSIS

We suppose that $V = \{3, 5\}$ is the weights set of the sensing nodes $ID = \{1, \ldots\ldots, 8\}$. Threshold authentication is defined as follows:

$$3x_1 + 5x_2 \geq 30.$$

The equation is divided by 15 on both sides:

$$\frac{x_1}{5} + \frac{x_2}{3} \geq 2.$$

Let $f(x) = 7 + 4x \bmod 13$, where $f(x) = 7$ is the shared key.

Let the primary identify $id_1, \ldots\ldots, id_c = 1, 2, 3, 4$, and we calculate that:

$$\begin{aligned} S_1 &= f(id_1) = 11. \\ S_2 &= f(id_2) = 2. \\ S_3 &= f(id_3) = 6. \\ S_4 &= f(id_4) = 10. \end{aligned}$$

Let $TV_1 = \{1, \ldots\ldots, 5\}$, $TV_2 = \{6, \ldots\ldots, 11\}$, and define the following polynomials:

$$\begin{aligned} f_1(x) &= 11 + \sum_{i=2}^{5} a_{1,i} x^{i-1}, \\ f_2(x) &= 2 + \sum_{i=2}^{10} a_{2,i} x^{i-1}, \\ f_3(x) &= 6 + \sum_{i=2}^{3} a_{3,i} x^{i-1}, \\ f_4(x) &= 10 + \sum_{i=2}^{6} a_{4,i} x^{i-1}, \end{aligned}$$

For each node $ID_i \in TV_1$, the key distributed to this node is $f_1(ID_i)f_2(ID_i)$.

For each node $ID_i \in TV_2$, the key distributed to this node is $f_3(ID_i)f_4(ID_i)$.

The core technique of the threshold signature procedure is the indirect recovery of the threshold key in the operation. The key recovery process for this scenario is illustrated as follows:

We assume that there are five nodes in TV_1 and three nodes in TV_2, respectively involved in the key recovery process. The key recovery process is described in the followings:

TABLE 3. Comparison our scheme with [50].

	Key distribution phase	Signature phase	Verification phase
Ours	$O(mE)$	$O(mV_0(k+1))$	E
[50]	$O(mE)$	$O(mE)$	$O(mE)$

First of all, the five nodes in TV_1 exploit Lagrangian multiplier method to recover the eleventh key of $f_1(x)$, then the three nodes in TV_2 take advantage of Lagrangian multiplier method to recover the key of $f_3(x)$. Finally, we use 6 and 11 to get $f(0) = 7$ by Lagrangian multiplier method.

5) VALIDITY

In Table 3 we compare the efficiency of the proposed scheme with the existing scheme, with m nodes participated in the signature and the average of M_i is M_0.

In the three phases mentioned before, only signature phase is completed by sensing nodes. E represents exponential operation of large numbers, k represents the threshold. Set $id_1, id_2, \ldots\ldots = 1, 2, \ldots\ldots$, that is, the primary identify is integer series beginning with 1, and $M_0(k+1) \ll E$.

We can see from table 3, the computation complexity of the proposed scheme, especially the signature phase performed by the sensing nodes, is much lower than that of the scheme [50], and it can match the computational power of the sensing nodes.

B. THRESHOLD AUTHENTICATION BASED ON HYBRID WEIGHTS

Consider the threshold equation of hybrid weights, which is donated as:

$$\frac{b_1 x_1 + q_1}{M_1} + \ldots\ldots + \frac{b_t x_t + q_t}{M_t} \geq k. \tag{12}$$

We suppose that $M_1, \ldots\ldots, M_t, b_1, \ldots\ldots, b_t, q_1, \ldots\ldots, q_t, k$ are positive integers, and $M_1, \ldots\ldots, M_t$ are inter-prime while $b_1, \ldots\ldots, b_t, k$ are inter-prime as well. The threshold equation of hybrid weights can be transformed into an integer weight threshold equation, which is donated as:

$$\begin{aligned} &b_1 M_2, \ldots, M_t x_1 + \ldots\ldots + b_t M_1, \ldots, M_{t-1} x_t \\ &\geq k M_1, \ldots, M_t - q_1 M_2, \ldots, M_t - \ldots\ldots - q_t M_1, \ldots, M_{t-1}. \end{aligned}$$

TV_i is a set of sensing nodes corresponding to x_i, and the size of this set is c_i.

1) KEY DISTRIBUTION

First of all, the key distribution manager KDS exploits the polynomial $f(x)$ defined on Z_q, which is donated as:

$$f(x) = sk + \sum_{i=1}^{k} a_{t-1} x^{t-1},$$

where $f(0) = sk$ is the shared key. KDS calculates and publish the public key $pk = g^{sk}$.

Denote

$$c = \sum_{i=1}^{t} \min\{\lfloor b_i(c_i + q_i)/M_i \rfloor, k\}.$$

Choose the primary identity $id_1, \ldots\ldots, id_c$, and generate the key fragments $S_1, \ldots\ldots, S_c$ in the following steps.

$$S_i = f(id_i), \quad 1 \le i \le c,$$

1) Then we denote

$$p_i = \sum_{j=1}^{i-1} \min\{\lfloor b_i(c_j + q_j)/M_j \rfloor, k\}, \quad 1 \le i \le t,$$
$$k_i = \min\{\lfloor b_i(c_i + q_i)/M_i \rfloor, k\},$$
$$\hat{k}_i = \min\{\lfloor (c_i + q_i)/M_i \rfloor, \lceil k/b_i \rceil\}.$$

2) The key distribution managers exploit the key fragments, which is donated as $S_{p_i+1}, \ldots\ldots, S_{p_i+k_i}$ to generate the key again, then delivers the fragments to the nodes in the TV_i in turn. The polynomial selected by the key distribution manager KDS is expressed as:

$$f_{p_i+1}(x) = S_{p_i+1} + a_{p_i+1,1}x + \ldots\ldots + a_{p_i+1,M_i-q_i-1}x^{M_i-q_i-1},$$
$$f_{p_i+2}(x) = S_{p_i+2} + a_{p_i+2,1}x + \ldots\ldots + a_{p_i+2,2M_i-q_i-1}x^{2M_i-q_i-1},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$f_{p_i+\hat{k}_i}(x) = S_{p_i+\hat{k}} + a_{p_i+\hat{k},1}x + \ldots\ldots + a_{p_i+\hat{k},\hat{k}_iM_i-q_i-1}x^{\hat{k}_iM_i-q_i-1}.$$

For each node ID_j^i in TV_i, we calculate

$$S_{p_i+1}^j = f_{p_i+1}(ID_j^i), S_{p_i+2}^j = f_{p_i+1}(ID_j^i + 1), \ldots\ldots\ldots\ldots\ldots,$$
$$S_{p_i+b_i}^j = f_{p_i+1}(ID_j^i + b_i - 1),$$
$$S_{p_i+b_i+1}^j = f_{p_i+2}(ID_j^i), S_{p_i+b_i+2}^j = f_{p_i+2}(ID_j^i + 1), \ldots\ldots\ldots\ldots\ldots,$$
$$S_{p_i+2b_i}^j = f_{p_i+2}(ID_j^i + b_i - 1),$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$S_{p_i+(c_i-1)b_i+1}^j = f_{p_i+\hat{k}}(ID_j^i), S_{p_i+(c_i-1)b_i+2}^j = f_{p_i+\hat{k}}(ID_j^i + 1), \ldots\ldots\ldots\ldots\ldots,$$
$$S_{p_i+k_i}^j = f_{p_i+\hat{k}}(ID_j^i + b_i - 1).$$

Deliver $S_{p_i+1}^j, \ldots\ldots, S_{p_i+\hat{k}_i}^j$ to the nodes ID_j^i.

For each node N_i ($i = 1, \ldots, m$), KDS selects a single-signed key sk_i, then we calculate $r_i = g^{sk_i}$, add (ID_i, r_i) to the public signature list and send sk_i to N_i by the secure channel. Different from the generation of shared key, a single-signed key sk is generated in each signature generating process. In order to reduce the burden in each sensing node beyond normal interaction, KDS uses mass generation or mass-produce techniques.

2) PROTOCOL DESIGN

Suppose that the sensing nodes $N_1, \ldots\ldots, N_{k_l}$ participate in the signature process, where the number of these sensing nodes in TV_i set is c_i'. H is the security-encoded hash function. Then we assume that λ_i is the Lagrangian multiplier corresponding to the primary identification $N_1, \ldots\ldots, N_{k_l}$ which we have mentioned before. A whole threshold authentication scheme includes the following steps.

1) As the number of nodes participating in signature procedure is c_i', for each node in TV_i, we select polynomials $f_{p_i+1}, .., f_{p_i+b_ic_i'/M_i}$ and calculate the following values:

The Lagrange multipliers corresponding to the first M_i nodes: $\lambda_{p_j+1,j}$ are the Lagrange multipliers corresponding to the $ID_1^i, \ldots\ldots, ID_{M_i}^i$, $\lambda_{p_i+2,j}$ are the Lagrange multipliers corresponding to $ID_1^i + 1, \ldots\ldots, ID_{M_i}^i + 1, \ldots \lambda_{p_i+b_i,j}$ are the Lagrange multipliers corresponding to $ID_1^i + b_i - 1, \ldots\ldots, ID_{M_i}^i + b_i - 1$.

The Lagrange multipliers corresponding to the first $2M_i$ nodes are expressed as follows: $\lambda_{p_i+b_i+1,j}$ are Lagrange multipliers corresponding to $ID_1^i, \ldots\ldots, ID_{2M_i}^i$, $\lambda_{p_i+b_i+2,j}$ are Lagrange multipliers corresponding to $ID_1^i + 1, \ldots\ldots,$ $ID_{2M_i}^i + 1, \ldots, \lambda_{p_i+2b_i,j}$ are Lagrange multipliers corresponding to $ID_1^i + b_i - 1, \ldots\ldots, ID_{2M_i}^i + b_i - 1$.

The Lagrange multipliers of all nodes participated in signature procedure in TV_i are expressed as follows: $\lambda_{p_i+(c'-1)b_i+1,j}$ are Lagrange multipliers corresponding to $ID_1^i, \ldots\ldots, ID_{c_i'}^i$, $.\lambda_{p_i+(c'-1)b_i+2,j}$ are Lagrange multipliers corresponding to $ID_1^i + 1, \ldots\ldots, ID_{c_i'}^i + 1, \ldots . \lambda_{p_i+c'b_i,j}$ are Lagrange multipliers corresponding to

$$ID_1^i + b_i - 1, \ldots\ldots, ID_{c_i'}^i + b_i - 1.$$

For each node ID_j^i in TV_i, we calculate that

$$s_{p_i+\lfloor b_i(j-1)/M_i \rfloor+1} = \lambda_{p_i+\lfloor b_i(j-1)/M_i \rfloor+1}\lambda_{p_i+\lfloor b_i(j-1)/M_i \rfloor+1,j} \times f_{p_i+\lfloor (j-1)/M_i \rfloor+1}(ID_j^i)H(m),$$
$$s_{p_i+\lfloor b_i(j-1)/M_i \rfloor+2} = \lambda_{p_i+\lfloor b_i(j-1)/M_i \rfloor+2}\lambda_{p_i+\lfloor b_i(j-1)/M_i \rfloor+2,j} \times f_{p_i+\lfloor (j-1)/M_i \rfloor+1}(ID_j^i + 1)H(m),$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$s_{p_i+\lfloor b_i(j-1)/M_i \rfloor+b_i} = \lambda_{p_i+\lfloor b_i(j-1)/M_i \rfloor+b_i}\lambda_{p_i+\lfloor b_i(j-1)/M_i \rfloor+b_i,j} \times f_{p_i+\lfloor (j-1)/M_i \rfloor+1}(ID_j^i + b_i - 1)H(m),$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$s_{p_i+\lfloor (c'-1)b_i/M_i \rfloor+1} = \lambda_{p_i+\lfloor (c'-1)b_i/M_i \rfloor+1}\lambda_{p_i+\lfloor (c'-1)b_i/M_i \rfloor+1} \times f_{p_i+\hat{k}_i}(ID_j^i)H(m),$$
$$s_{p_i+\lfloor (c'-1)b_i/M_i \rfloor+2} = \lambda_{p_i+\lfloor (c'-1)b_i/M_i \rfloor+2}\lambda_{p_i+\lfloor (c'-1)b_i/M_i \rfloor+2} \times f_{p_i+\hat{k}_i}(ID_j^i + 1)H(m),$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$s_{p_i+k_i}^j = \lambda_{p_i+k_i}\lambda_{p_i+k_i,j}f_{p_i+\hat{k}_i}(ID_j^i + b_i - 1)H(m) - sk_j^i.$$

$$
\begin{aligned}
r &= \prod_{i=1}^{k_l} r_i, \\
g^s r &= g^{\sum\limits_{V_i}\sum\limits_{}\sum\limits_{}\lambda_{p_i+\lfloor b_i(j-1)/M_i\rfloor+z}\lambda_{p_i+\lfloor b_i(j-1)/M_i\rfloor+z,j} f_{p_i+\lfloor (j-1)/M_i\rfloor+z/b_i}(ID_j^i+z/b_i)H(m)} \; g^{-\sum\limits_{i=1}^{k_l} sk_i} \; g^{-\sum\limits_{i=1}^{k_l} sk_i} \\
&= g^{\sum\limits_{i=1}^{k}\lambda_i f(id_i)H(m)} \\
&= g^{f(0)H(m)} = pk^{H(m)}.
\end{aligned}
$$

Assume that sk_j^i is the single-signed key of ID_j^i, and for node ID_j^i we calculate

$$s_j^i = \sum_{p_i+\lfloor b_i(j-1)/M_i\rfloor+1}^{p_i+k_i} s_z.$$

Then node ID_j^i will send the signature result s_j^i to Sink node.

2) Sink node collects all signature fragments s_i^j of all nodes, combines them with m and sends this final signature to the challenger party.
3) The challenger the sum of all signature fragments, donated as $s = \sum s_j^i$ and verifies this process as r and $g^s r$, shown at the top of this page, which can be also indicated as $g^s r = pk^{H(m)}$.

3) FORMAL SECURITY ANALYSIS

It is obvious that any $x_1, \ldots\ldots, x_t$ satisfying the equation

$$\left\lfloor \frac{b_1(x_1+q_1)}{M_1} \right\rfloor + \ldots\ldots + \left\lfloor \frac{b_t(x_t+q_t)}{M_t} \right\rfloor \geq k,$$

can get n_c values of $f(x)$, so the signature process can complete successfully. According to the characteristics of key distribution, signature combinations that do not satisfy the threshold equation will not have enough key fragments. Moreover, they cannot be encrypted, signed and execute other steps. Therefore, this scheme can resist any collusion attack that is less than the threshold value.

4) CASE ANALYSIS

We consider the following threshold equation with hybrid weight

$$\frac{2x_1}{3} + \frac{3x_2+1}{5} + \frac{5x_2+1}{7} \geq 2,$$

Whose corresponding integer equation is:

$$70x_1 + 63x_2 + 75x_3 \geq 174$$

Denote

$$f(x) = 7 + 4x \bmod 13,$$

where $f(x) = 7$ is the shared key.

Let the primary identify $id_1, \ldots\ldots, id_c = 1, 2, 3, 4, 5, 6$, and then calculate

$$S_1 = f(id_1) = 11, \quad S_2 = f(id_2) = 2,$$
$$S_3 = f(id_3) = 6, \quad S_4 = f(id_4) = 10,$$
$$S_5 = f(id_5) = 1, \quad S_6 = f(id_6) = 5.$$

Let $TV_1 = \{1, \ldots\ldots, 3\}$, $TV_2 = \{4 \ldots\ldots, 9\}$, $TV_3 = \{10, \ldots\ldots, 16\}$, and define the polynomials as follows:

$$
\begin{aligned}
f_1(x) &= 11 + \sum_{i=2}^{3} a_{1,i}x^{i-1}, \\
f_2(x) &= 2 + \sum_{i=2}^{6} a_{2,i}x^{i-1}, \\
f_3(x) &= 6 + \sum_{i=2}^{4} a_{3,i}x^{i-1}, \\
f_4(x) &= 10 + \sum_{i=2}^{9} a_{4,i}x^{i-1}, \\
f_5(x) &= 1 + \sum_{i=2}^{6} a_{5,i}x^{i-1}, \\
f_6(x) &= 5 + \sum_{i=2}^{13} a_{6,i}x^{i-1}.
\end{aligned}
$$

The keys distributed to each node $ID_i \in TV_1$ are expressed as:

$$f_1(ID_i), \quad f_1(ID_i+1).$$
$$f_2(ID_i), \quad f_2(ID_i+1).$$

The keys distributed to each node $ID_i \in TV_2$ are expressed as:

$$f_3(ID_i), \quad f_3(ID_i+1), \quad f_3(ID_i+2),$$
$$f_4(ID_i), \quad f_4(ID_i+1), \quad f_4(ID_i+2).$$

The keys distributed to each node $ID_i \in TV_3$, are expressed as:

$$f_5(ID_i), \quad f_5(ID_i+1), \quad f_5(ID_i+2), \quad f_5(ID_i+3),$$
$$f_5(ID_i+4), \quad f_6(ID_i), \quad f_6(ID_i+1), \quad f_6(ID_i+2),$$
$$f_6(ID_i+3), \quad f_6(ID_i+4),$$

The core technique of the threshold signature procedure is the indirect recovery of the threshold key during operations. The key recovery process for this scenario is illustrated as follows:

We assume that there are 2 nodes in TV_1 as well as 2 nodes in TV_2 involved in the key recovery process with no node in TV_3 involved in the same process. The key recovery is described in the following steps:

First of all, the 2 nodes in TV_1 exploit Lagrangian multiplier method to recover the eleventh key of $f_1(x)$, then the 2 nodes in TV_2 take advantage of Lagrangian multiplier method to recover the sixth key of $f_3(x)$. Finally, we use 6 and 11 to get $f(0) = 7$ by Lagrangian multiplier method.

For other solutions of the same equation such as (2, 0, 2) and (0, 2, 2). The recovery method is same as we mentioned before.

5) VALIDITY

In Table 4 we compare the efficiency of the proposed scheme with the existing scheme, with m nodes participated in the signature and the average of b_iM_i is M_b.

TABLE 4. Comparison our scheme with [50].

	Key distribution phase	Signature phase	Verification phase
Ours	$O(mE)$	$O(mM_b(k+1))$	E
[50]	$O(mE)$	$O(mE)$	$O(mE)$

In the three phases mentioned before, only signature phase is completed by sensing nodes. E represents exponential operation of large numbers, k represents the threshold. Set $id_1, id_2, \ldots\ldots = 1, 2, \ldots\ldots$, that is, the primary identify is integer series beginning with 1, and $M_b(k+1) \ll E$.

We can see from table 3, the computation complexity of the proposed scheme, especially the signature phase performed by the sensing nodes, is much lower than that of the scheme [50], and it can match the computational power of the sensing nodes.

VI. CONCLUSION AND FUTURE WORK

Due to the higher complexity and the work by the equality of members, the existing threshold authentication schemes cannot be suitable to the circumstance with various sensing nodes. In this paper, we firstly propose key distribution scheme in sensing networks based on symmetric operators. By the permutation function, the proposed scheme could generate feature sequences to improve the efficiency of key distribution in the sensing networks. In addition, we propose a weighed threshold authentication scheme based on weight value level. The higher the weight of the sensing node is, the greater the role of the node plays, and vice versa. The proposed scheme only uses finite multiplicative operations, which can match the computing power of the sensing nodes and reduce the burden of each sensing nodes.

In future, we will mainly focus on the improvement of threshold authentication scheme based on weights to be adapted to different applications of IoT.

REFERENCES

[1] R. Roman, J. Zhou, and J. Lopez, "On the features and challenges of security and privacy in distributed internet of things," *Comput. Netw.*, vol. 57, no. 10, pp. 2266–2279, Jul. 2013.

[2] B. Gong, Y. Zhang, and Y. B. Wang, "A remote attestation mechanism for the sensing layer nodes of the Internet of Things," *Future Gener. Comput. Syst.*, vol. 78, pp. 867–886, Jan. 2018.

[3] N. Xiong *et al.*, "Comparative analysis of quality of service and memory usage for adaptive failure detectors in healthcare systems," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, pp. 495–509, May 2009.

[4] N. Xiong, X. Jia, L. T. Yang, A. V. Vasilakos, Y. Li, and Y. Pan, "A distributed efficient flow control scheme for multirate multicast Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 9, pp. 1254–1266, Sep. 2010.

[5] N. Xiong *et al.*, "A novel self-tuning feedback controller for active queue management supporting TCP flows," *Inf. Sci.*, vol. 180, no. 11, pp. 2249–2263, Jun. 2010.

[6] S. J. Wen, C. H. Huang, X. Chen, J. H. Ma, N. X. Xiong, and Z. P. Li, "Energy-efficient and delay-aware distributed routing with cooperative transmission for Internet of Things," *J. Parallel Distrib. Comput.*, vol. 118, pp. 46–56, Aug. 2018.

[7] L. Wan, L. J. Wei, N. X. Xiong, J. J. Yuan, and J. C. Xiong, "Pareto optimization for the two-agent scheduling problems with linear non-increasing deterioration based on Internet of Things," *Future Gener. Comp. Syst.*, vol. 76, pp. 293–300, Nov. 2017.

[8] Y. Fang, Q. Chen, N. X. Xiong, D. Zhao, and J. Wang, "RGCA: A reliable GPU cluster architecture for large-scale Internet of Things computing based on effective performance-energy optimization," *Sensors*, vol. 17, no. 8, p. 1799, 2017.

[9] M. Yi, Q. Chen, and N. X. Xiong, "An effective massive sensor Network data access scheme based on topology control for the Internet of Things," *Sensors*, vol. 16, no. 11, p. 1846, 2016.

[10] A. V. Vinel, W. S. E. Chen, N. X. Xiong, S. M. Rho, N. Chilamkurti, and A. V. Vasilakos, "Enabling wireless communication and networking technologies for the internet of things," *IEEE Wireless Commun.*, vol. 23, no. 5, pp. 8–9, Oct. 2016.

[11] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security protocols for sensor Networks," in *Proc. 7th Annual Int. Conf. Mobile Comput. Netw.*, Aug. 2001, pp. 189–199.

[12] Q. Huang, J. Cukier, H. Kobayashi, B. Liu, and J. Y. Zhang, "Fast authenticated key establishment protocols for self-organizing sensor Networks," in *Proc. 2nd ACM Int. Conf. Aware Netw. Appl.*, Sep. 2003, pp. 141–150.

[13] R. Watro, D. Kong, S. F. Cuti, C. Gardiner, C. Lynn, and P. Kruus, "TinyPK: Securing sensor Networks with public key technology," in *Proc. 2nd ACM Workshop Secur. Ad Hoc Sensor Netw.*, Oct. 2004, p. 59- 64.

[14] D. Nyang and A. Mohaisen, "*Cooperative Public Key Authentication Protocol in Wireless Sensor Network*, New York, NY, USA: Springer, 2006, pp. 864–873.

[15] K. Bauer and H. Lee, "A distributed authentication scheme for a wireless sensing system," in *Proc. 2nd Int. Workshop Networked Sensing Syst.*, Jun. 2005, pp. 210–215.

[16] H. R. Tseng, R. H. Jan, W. Yang, "An improved dynamic user authentication scheme for aware Network," in *Proc. IEEE Global Telecommun. Conf.*, Aug. 2007, pp. 986–990.

[17] C. Jiang, B. Li, and H. X. Xu, "An efficient scheme for user authentication in aware Network," in *Proc. 21st Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Apr. 2007, pp. 438–442.

[18] G. Q. Xu, Y. Zhang, A. K. Sangaiah, X. H. Li, A. Castiglione, and X. Zheng, "CSP-E2: An abuse-free contract signing protocol with low-storage TTP for energy-efficient electronic transaction ecosystems," *Inf. Sci.*, vol. 476, pp. 505–515, Feb. 2019. doi: 10.1016/j.ins.2018.05.022.

[19] G. Q. Xu, J. Liu, Y. R. Lu, X. J. Zeng, Y. Zhang, and X. M. Li, "A novel efficient MAKA protocol with desynchronization for anonymous roaming service in global mobility Networks," *J. Netw. Comput. Appl.*, vol. 107, pp. 83–92, Apr. 2018.

[20] R. Wang, G. Q. Xu, B. Liu, Y. Cao, and X. Li, "Flow watermarking for antinoise and multistream tracing in anonymous Networks," *IEEE MultiMedia*, Vol.24, no. 4, pp. 38–47, Dec. 2017.

[21] R. Wang, G. Xu, X. Zeng, X. Li, and Z. Feng, "TT-XSS: A novel taint tracking based dynamic detection framework for DOM cross-site scripting," *J. Parallel Distrib. Comput.*, vol. 118, pp. 100–106, Aug. 2018.

[22] G. Q. Xu, Z. Y. Feng, H. B. Wu, and D. X. Zhao, "Swift trust in virtual temporary system: A model based on dempster-shafer theory of belief functions," *Int. J. Electron. Commerce*, Vol. 12, no. 1, pp. 93–127, 2007.

[23] X. Zeng, G. Xu, X. Zheng, Y. Xiang, and W. Zhou, "E-AUA: An efficient anonymous user authentication protocol for mobile IoT," *IEEE Internet Things J.*, 2018. doi: 10.1109/JIOT.2018.2847447.
[24] G. Q. Xu *et al.*, "An algorithm on fairness verification of mobile sink routing in wireless sensor Network," *Pers. Ubiquitous Comput.*, vol. 17, no. 5, pp. 851–864, Jun. 2013.
[25] A. Karati, S. K. H. Islam, and M. Karuppiah, "Provably secure and lightweight certificateless signature scheme for IIoT environments," *IEEE Trans. Ind. Inform.*, vol. 14, no. 8, pp. 3701–3711, Aug. 2018.
[26] H. Taha and E. Alsusa, "Secret key exchange and authentication via randomized spatial modulation and phase shifting," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2165–2177, Mar. 2018.
[27] H. Tan, D. Choi, P. Kim, S. Pan, and I. Chung, "Comments on 'dual authentication and key management techniques for secure data transmission in vehicular ad hoc Networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2149–2151, Jul. 2017.
[28] K. A. Shim, "Comments on 'A cross-layer approach to privacy-preserving authentication in WAVE-enabled VANETs' by Biswas and Mišic," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10588–10589, Nov. 2017.
[29] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated Key establishment protocol for smart home environment," *IEEE Trans. Dependable Secure Comput.*, 2017.
[30] Y. L. Yao, L. T. Yang, and N. X. Xiong, "Anonymity-based privacy-preserving data reporting for participatory sensing," *IEEE Internet Things J.*, vol. 2, no. 5, pp. 381–390, Oct. 2015.
[31] D. R. Raymond and S. F. Midkiff, "Denial-of-service in wireless sensor Networks: Attacks and defenses," *Pervasive Comput.*, vol. 7, no. 1, pp. 74–81, Mar. 2008.
[32] M. Luk, G. Mezzour, A. Perrig, and V. Gligor, "MiniSec: A secure sensor Network communication architecture," in *Proc. 6th Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2007, pp. 479–488.
[33] S. A. Chaudhry, A. Albeshri, N. X. Xiong, C. H. Lee, and T. Shon, "A privacy preserving authentication scheme for roaming in ubiquitous networks," *Cluster Comput.*, vol. 20, no. 2, pp. 1223–1236, Jun. 2017.
[34] Y. Lu, S. Q. Wu, Z. J. Fang, N. X. Xiong, S. Yoon, and D. S. Park, "Exploring finger vein based personal authentication for secure IoT," *Future Gener. Comput. Syst.*, vol. 77, pp. 149–160, Dec. 2017.
[35] J. Wang, N. X. Xiong, J. H. Wang, and W. C. Yeh, "A compact ciphertext-policy attribute-based encryption scheme for the information-centric Internet of Things," *IEEE Access*, vol. 6, pp. 63513–63526, 2018.
[36] V. Gupta *et al.*, "Sizzle: A standards-based end-to-end security architecture for the embedded internet," *Pervasive Mobile Comput.*, vol. 1, pp. 425–445, Aug. 2005.
[37] W. Hu, H. Tan, P. Corke, W. C. Shih, and S. Jha, "Toward trusted wireless sensor Networks," *ACM Trans. Sensor Netw.*, vol. 7, pp. 5:1–5:25, Aug. 2010.
[38] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor Networks," in *Proc. Symp. Secur. Privacy*, May 2003, pp. 197–213.
[39] W. Jung, S. Hong, M. Ha, Y. J. Kim, and D. Kim, "SSL-based lightweight security of IP-based wireless sensor networks," in *Proc. Int. Conf. Adv. Inf. Netw. Appl. Workshops*, May 2009, pp. 1112–1117.
[40] T. Kothmayr, C. Schmitt, W. Hu, M. Brünig, and G. Carle, "DTLS based security and two-way authentication for the Internet of Things. *Ad Hoc Netw.*, vol. 11, no. 8, pp. 2710–2723, Nov. 2013.
[41] Y. Desmedt and Y. Frankel, *Threshold Cryptosystems*, vol. 435, Berlin, Germany: Springer, 1990, pp. 307–315.
[42] Y. Desmedt and Y. Frankel, *Shared Generation of Authenticators and Signatures*. Berlin, Germany: Springer, 1992, pp. 457–469.
[43] A. De Santis, Y. Desmedt, and Y. Frankel, "How to share a function securely," in *Proc. 26th ACM Symp. Theory Comput.* Santa Fe, Mexico, May 1994, pp. 522–533.
[44] R. Gennaro, S. Jareki, and H. Krawczyk, "Robust theshold DSS signatures," in *Advances in Cryptology—EUROCRYPT*, vol. 96. New York, NY, USA: Springer, 1996, pp. 354–371.
[45] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 3, pp. 20–613, Nov. 1979.
[46] Y. Desmedt and Y. Frankel, "Shared generation of authenticators," in *Proc. Annu. Int. Cryptol. Conf.*, 1991, pp. 457–469.
[47] C. T. Wang, C. H. Lin, and C. C. Chang, "Threshold signature schemes with traceable signers in group communications," *Comput. Commun.*, vol. 21, no. 8, pp. 771–776, Jun. 1998.
[48] T. P. Pedersen, "A threshold cryptosystem without a trusted party," in *Proc. Workshop Theory Appl. Cryptograph. Techn.*, May 1991, pp. 522–526.
[49] W. D. Chen and D. G. Feng, "A group of threshold group-signature schemes with privilege subsets," *J. Softw.*, vol. 16, no. 3, pp. 1289–1295, 2005.
[50] C. D. Constantin and L. T. Ferucio, "Distributive weighted threshold secret sharing schemes," *Inf. Sci.*, vol. 339, pp. 85–97, Apr. 2016.
[51] C. Blundo, A.D. Santis, A. Herzberg, and S. Kutten, "Perfectly-secure key distribution for dynamic conferences," *Adv. Cryptol., Inf. Comput.*, vol. 146, no. 1, pp. 1–23, Oct. 1993.

ZHENHU NING received the Ph.D. degree in computer sciences from the Beijing University of Technology, Beijing, in 2016, where he is currently with the Faculty of Information Technology. His research interests include the field of security IOT, including security wireless sensor networks, security sensing data transmission and security computing environment of the sensing node, the field of security cloud storage, security cloud privacy, malicious code detection, machine learning, system optimization and control, and practical partial differential equations.

GUANGQUAN XU (M'18) received the Ph.D. degree from Tianjin University, China, in 2008, where he is currently a Full Professor with the Tianjin Key Laboratory of Advanced Networking, College of Intelligence and Computing. His research interests include cyber security and trust management. He is a member of the CCF.

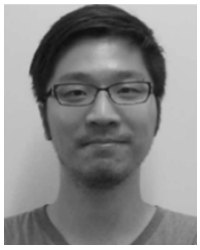
NAIXUE XIONG received the Ph.D. degree in sensor system engineering from Wuhan University and the Ph.D. degree in dependable sensor networks from the Japan Advanced Institute of Science and Technology. He is currently a Professor with the College of Intelligence and Computing, Tianjin University, China. Before he joined Tianjin University, he was with Northeastern State University, Georgia State University, Wentworth Technology Institution, and Colorado Technical University (Full Professor about 5 years) about 10 years. His research interests include cloud computing, security and dependability, parallel and distributed computing, networks, and optimization theory.

YONGLI YANG is currently pursuing the Ph.D. degree with the Faculty of Information Technology, Beijing University of Technology, Beijing. Her research interests include recommendation systems, machine learning, swarm intelligence, and quantum encryption.

CHANGXIANG SHEN is currently an Academician of the Chinese Academy of Engineering. His research areas include computer information systems, cryptographic engineering, information security architecture, system software security (security operation systems and security database), and network security.

EMMANOUIL PANAOUSIS received the B.Sc. degree in informatics and telecommunications from the University of Athens, Greece, in 2006, and the M.Sc. degree in computer science from the Athens University of Economics and Business, Greece, in 2008, and the Ph.D. degree in mobile communications security from Kingston University London, U.K., in 2012. He was a Senior Lecturer of cyber security and privacy with the University of Brighton, an Invited Researcher with Imperial College London, a Postdoctoral Researcher with the Queen Mary University of London, and a Research and Development Consultant with Ubitech Technologies Ltd., Surrey Research Park. He is currently a Lecturer (Assistant Professor) with the University of Surrey, U.K. He is a member of the Surrey Centre for Cyber Security, a GCHQ, which is a recognized U.K. Academic Centre of Excellence in Cyber Security Research. His research interests include the fields of cyber security, privacy, and algorithmic decision making.

HAO WANG received the B.Eng. and Ph.D. degrees in computer science and engineering. He is currently an Associate Professor with the Norwegian University of Science and Technology, Norway. His research interests include big data analytics, the industrial Internet of Things, high-performance computing, safety-critical systems, and communication security. He has published over 80 papers in reputed international journals and conferences. He has served as the TPC Co-Chair of the IEEE DataCom 2015, the IEEE CIT 2017, and ES 2017. He is a reviewer of journals such as the IEEE TKDE, TII, TBD, TETC, T-IFS, IoTJ, and ACM TOMM. He is a member of the IEEE IES Technical Committee on Industrial Informatics.



KAITAI LIANG received the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, in 2014. He is currently an Assistant Professor with the Department of Computer Science, University of Surrey, U.K. His research interests include applied cryptography and information security, in particular, encryption, blockchain, post-quantum crypto, privacy enhancing technology, and security in cloud computing.