A Secure Random Key Distribution Scheme Against Node Replication Attacks in Industrial Wireless Sensor Systems

Longpeng Li , Guangquan Xu*, *Member, IEEE,* Litao Jiao , Xiaotong Li , Hao Wang , *Member, IEEE,* Jing Hu , Hequn Xian , Wenjuan Lian , Honghao Gao , *Member, IEEE,*

.

***Abstract*—With the wide deployment of wireless sensor networks in smart industrial systems, lots of unauthorized attacking from the adversary are greatly threatening the security and privacy of the entire industrial systems, of which node replication attacks can hardly be defended since it is conducted in the physical layer. To solve this problem, we propose a secure random key distribution scheme, called SRKD, which provides a new method for the defense against the attack. Specifically, we combine a localized algorithm with a voting mechanism to support the detection and revocation of malicious nodes. We further change the meaning of the parameter s to help prevent the replication attack. Furthermore, the experimental results show that the detection ratio of replicate nodes exceeds 90% when the number of network nodes reaches 200, which demonstrates the security and effectiveness of our scheme. Compared with existing state-of-the-art schemes, SRKD also has good storage and communication efficiency.**

***Index Terms*—Industrial IoT, wireless sensor system, random key distribution, node replication attack**

I. Introduction

A Wireless sensor system (WSS) has been increasingly used in industrial Internet of Things (IoT) applications to date [1]–[3], such as the network of vehicles, the smart grid [4] and intelligent manufacturing [5], [6]. The WSS is a type of distributed, multihop and self-configuring sensor system that is equipped with an enormous number of sensor nodes [7], [8]. The sensors, which are limited in power, storage and resources in deployed systems, transfer data via dint of the wireless signal [9]. Since their deployment in valuable industrial applications, WSSs may attract system attackers to gain "benefits" from compromising the whole system or some specified system nodes via various types of attacks [10], e.g., hardware tampering, malicious message injection and node replication attacks. These attacks may incur massive economic losses to industries and even threaten the safety and stability of the local community. It is of extreme importance to protect WSS applications from information breaches and security threats [11].

Motivation: To establish resilience against node capture and information eavesdropping, a key management scheme is essentially employed in the context of the WSS. The random key distribution scheme has been studied as one of the most effective secure key bootstrapping approaches for WSS applications. A random key distribution scheme is the probability-based key management mechanism such that each node preloads with a set of keys selected from the key pool. If the neighbor nodes have a common key, they regard the common key as the pairwise key and further use it to establish a secure communication channel/link. The state-of-the-art random key distribution schemes are described in the next section.

Unfortunately, these state-of-the-art schemes have yet to hold against node replication attack so that the data flow and honest nodes may suffer from a high risk of being compromised. For example, nodes may be compromised by a system attacker who impersonates the captured nodes to fabricate replicas and further deploy the malicious replicas into the system to engage in evildoing, e.g., insider attacks [12], which may cause instability of the whole system. In practice, it may be challenging to design an effective defense mechanism to this attack because the credentials of these malicious replicas are a replication of the compromised honest nodes; therefore, the replicas may be seen as "legitimate".

Our contributions: This paper proposes a new scheme called SRKD. The innovation of our proposed SRKD is described as follows. We propose a novel secure method to deal with replicate nodes. Firstly, we combine the node replication attack

This work is partially sponsored by the State key development program of China(No. 2018YFB0804402), and National Science Foundation of China (No. 61572355, U1736115)

Longpeng Li, Tianjin Key Laboratory of Advanced Networking (TANK), College of Intelligence and Computing, Tianjin University, China, 300350 (Email: lilongpengllp@126.com).

Guangquan Xu(Corresponding author), Tianjin Key Laboratory of Advanced Networking (TANK), College of Intelligence and Computing, Tianjin University, Tianjin China; Qingdao Huanghai University, No. 1145, Linghai Road, Huangdao District, Qingdao City, Shandong Province, China (Email: losin@tju.edu.cn).

Litao Jiao, Qingdao Huanghai University, No. 1145, Linghai Road, Huangdao District, Qingdao City, Shandong Province, China (Email: jiaolitao_11@163.com).

Xiaotong Li, Tianjin Key Laboratory of Advanced Networking (TANK), College of Intelligence and Computing, Tianjin University, Tianjin China, 300350 (Email: lixiaotong@tju.edu.cn).

Hao Wang, Department of Computer Science, Norwegian University of Science and Technology, Norway (Email: hawa@ntnu.no).

Jing Hu, College of Intelligence and Computing, Tianjin University, Tianjin China, 300350 (Email: mavis_huhu@tju.edu.cn).

Hequn Xian, College of Computer Science and Technology, Qingdao University, China (Email: xianhq@qdu.edu.cn).

Wenjuan Lian, College of Computer Science and Engineering, Shandong University of Science and Technology, China (Email: skd991457@sdust.edu.cn).

Honghao Gao, Computing Center,Shanghai University, Shanghai, 200444, China (Email: gaohonghao@shu.edu.cn).

detection called Efficient Distributed Detection (EDD) algorithm [12] with a voting mechanism to provide detection and replicas revocation. Then, We change the semantic meaning of the parameter s to further help eradicate the replication attack. In this paper, our s is the maximum number of starting keys that can be used to establish a communication link. If the number of shared starting keys of two nodes is equal or greater than s, the link will not be established because there would be a high probability of node replication. After that, We also introduce into SRKD a message recovery mechanism called full message recovery ID-based signature (MR-IBS), proposed by Shim et al. [13], to reduce the network bandwidth overhead caused by the delivery of voting messages to a base station.

Compared with previous random key distribution schemes, our new scheme contributes great improvements in terms of security and efficiency. It also performs well in comparison with other replica detection schemes. The experimental results demonstrate that our SRKD approach can effectively hold against the node replication attack while maintaining secure key bootstrapping.

Organization: The remainder of the paper is organized as follows. In Section II, we present related works. In Section III, we present the system model in SRKD. In Section IV, we introduce our approach and the proposed scheme. In Section V, we evaluate our scheme. In Section VI, we provide an industrial application. Finally, in Section VII, we draw conclusions and remarks regarding future work.

II. Related Work

A. Random Key Distribution Scheme

The first random key distribution was proposed by Eschenauer and Gligor (EG) [14]. Unfortunately, EG is no longer secure. A subsequent evolution of EG is the q-Composite (QC) proposed by Chan et al. [15]. In QC, two neighboring nodes can establish a link and build up the communication only if they share at least q starting keys, and they generate a new pairwise key by performing a hash function on the concatenation of all the shared keys. QC has a higher level of security compared to EG even if a small number of nodes are compromised. However, a larger amount of memory is required in QC, which is the main bottleneck in practical use. An evolution of QC is the q-s-Composite (QSC), which is proposed by Gandino et al. [16]. An upper bound parameter s is used to limit the number of keys, while a light generation technique of the pairwise key based on the wise XOR is proposed to reduce the complexity of calculation. However, QSC fails to hold against the node replication attack.

B. Defense Against Node Replication Attack

In the research line of the defense against node replication attacks, most distributed detection protocols make use of the witness-finding method to detect replicate nodes [17]. Note that these approaches generally are based on the assumption that a sensor node should broadcast a signed location message to its neighbor nodes [18]. For instance, the witnesses are determined randomly in the scheme of randomized multicast (RM) and the scheme of line-selected multicast (LSM) proposed in [17]. Randomized, efficient, and distributed (RED) [19] is based on the same concept, but it enjoys a lighter communication cost. Parallel multiple probabilistic cells (P-MPC) and single deterministic cell (SDC) [18] are also witness-based schemes. Regarding the scheme of [20], it is based on the double ruling.

Yu et al. propose the first distributed detection algorithm against the node replication attack based on a simple challenge-response method in a mobile WSS. However, the algorithm does not effectively defend against collusive replicas.

To address the limitation, Yu et al. [12] design two localized algorithms called XED and EDD based on the strategy of challenge-response and encounter-number. They aim to help to significantly reduce the communication overhead.

III. System Model and Definition

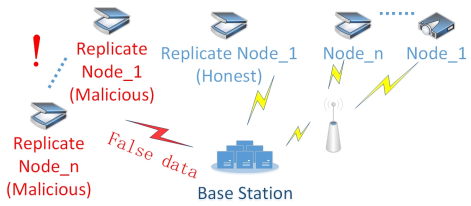


Fig. 1. System model.

In industrial wireless sensor systems, existing devices are often replicated for data backup, function expansion, and other situations. A sensor system model is shown in Fig. 1. The system replicates some honest nodes to enhance the processing power of the sensor cluster and expand the sensing range. Unfortunately, an adversary can use the information from captured honest nodes to create malicious replication nodes. The malicious replicas can transmit false data to the system, which may affect the stability of the system and even cause tremendous damage.

Attack Model: In SRKD, after sensor deployment, nodes can be compromised immediately by the opponent. The network adversary can eavesdrop on all the data transmitted. It can even inject bogus data into the system to consume the resources. The adversary cannot obtain the starting keys (k_i), but it knows the identifier ID_{k_i} of the starting keys and the identifier $ID_{K_{share}}$ of the pairwise keys. The adversary cannot change the data in the compromised nodes. However, it has all the legitimate credentials and starting keys from those compromised nodes. Then, it can replicate malicious nodes with the information of compromised nodes. Although both benign and malicious replicas may replicate the information from the same honest node, there are obvious differences between the benign and malicious replicas. Benign replicas are used for formal purposes and authorized by the system. However, malicious replicas are unauthorized nodes that threaten the security of the system. They bring with them abnormal data flow and behavior patterns.

IV. Our Proposed Scheme

A. The Process of SRKD

To provide a mechanism against the node replication attack for random key distribution, we propose a novel scheme called SRKD. The workflow and implementation steps of our scheme are shown in Algorithm 1. Details of the algorithm will be expanded in the later parts of this section.

B. Notations

To describe the proposed scheme, we first introduce some parameters that will be used later in the paper.

- n: the number of nodes in the network.
- $n^{'}$: the number of neighbouring nodes.
- e: the expected number of neighbour nodes for the node within communication range.
- p: the number of starting keys in the key pool.
- r: the number of starting keys of each ring that a node has.
- q: the minimum number of shared keys that two neighbour nodes can establish a link.
- s: the maximum number of shared keys that two neighbour nodes can establish a link (Open interval).
- T_{vote}: the voting threshold that a node can be thought as a replication node.
- d: the expected number of links that a node can establish in key-setup phase.
- pro: the probability that two neighbouring nodes can establish a secure link during the key-setup phase.

Algorithm 1 : Execution Process of SRKD

1. SRKD: Execute the first deployment
2. Execute the keys distribution phase
3. Execute the keys establishment phase
4. Detect the node replication attack
5. **while** Node replication attacks exist in the system (the detection phase) **do**
6. Execute the revocation of replicas (the revocation phase)
7. **end while**
8. Execute key updating periodically
9. Begin the next deployment
10. Execute the prevention (the prevention phase)
11. **if** The shared starting keys in nodes $>= s$ **then**
12. Revoke those nodes
13. **else**
14. Execute the next keys distribution and establishment
15. **end if**

C. Random Keys Distribution in SRKD

1) Keys Distribution Phase: Each key has its identity ID_{k_i}. The scheme of SRKD selects p keys randomly from the key space to establish the key pool before the deployment of the network. Each node further randomly chooses r keys in the ring as the starting keys. Additionally, the base station will distribute a unique identifier ID_i for each node so that the node stores the ring and their ID_i.

2) Keys Establishment Phase: In this phase, each node broadcasts a handshake message periodically. Fig. 2 shows the storage contents of the ring in different nodes. Fig. 3 to Fig. 4 give a brief description of key establishment and the final state. If a node sends the handshake message, then we call it initiator. The handshake message contains the identity (ID_i) of the initiator and all the ID_{k_i} of the keys (K_i) in the ring of the initiator. A node called receiver receives the handshake message and checks those shared starting keys in the received ID_{k_i}.

If the number of shared keys is less than q, then it cannot establish a link between the two nodes and then the handshake will be stopped by the receiver. If the number of the shared keys is between q and s, then the receiver records the ID_i of the initiator and the ID_{k_i} of the shared keys. SRKD does not store the pairwise key since the pairwise key is generated every time before being used. The pairwise key is calculated by the bitwise XOR of the shared keys ($K_{share} = K_1 \oplus K_2 \oplus \ldots K_i \oplus \ldots \oplus K_n$. The identifier of it is $ID_{K_{share}}$). Next, the receiver sends an acknowledge message to the initiator. This message contains the identity of the receiver and ID_{k_i} of the shared starting keys. A MAC executed by the pairwise key is used to authenticate the acknowledge message. The initiator receives and checks the MAC and then calculates the pairwise key. It will store the ID_{k_i} of the shared keys from the receiver with the identifiers (the position in the array) if the calculation results are correct. The information can be used to calculate the pairwise key when it is needed.

However, when the first set of nodes has been deployed, the key establishment phase is different from QSC. In this situation, SRKD takes parameter s into consideration. If the number of shared keys that the receiver found is greater than or equal to s, then it will not establish the link between the two nodes. Because all keys are picked randomly from a large key pool and every node also picks the ring randomly from the key pool, it is less likely that two nodes will have more than s common keys. Thus, we believe that the node replication attack occurs. It can relieve the harm that replicas may inflict on the network to a certain extent and reduce the cost of the node revocation phase.

We also introduce a message recovery mechanism in SRKD to reduce the network bandwidth overhead. For more details, please refer to [21].

D. The Detection Phase

We make an adjustment based on the detection algorithm of EDD proposed by Yu et al. [12] in the detection phase.

1) Detection of The Node Replication Attack (EDD-Off-line-Step): In Algorithm 2, EDD calculates μ_1, μ_2,σ_1^2, σ_2^2, Y_1 and Y_2 to obtain the threshold of ψ such that a node can be considered as a replica.

Algorithm 2 : EDD-Off-Line-Step

1. set $T = 1$, $\mathcal{B}^{(\mu)} = \phi$, $\mu \in [1, n]$
2. set $\mathcal{L}^{(\mu)}[i] = 0$, $1 \leq i \leq n$,$\mu \in [1, n]$
3. **while** $Y_1 >= Y_2$ **do**
4. $T = T + 1$
5. calculate μ_1, μ_2, σ_1^2, σ_2^2
6. set $Y_1 = \mu_1 + 3\sigma_1$ and $Y_2 = \mu_2 - 3\sigma_2$
7. **end while**
8. set $\psi = \frac{Y_2 - Y_1}{2}$

The term $L^{(u)}$ records the count that node u encounters neighboring nodes in the network during each time interval of T. For $B^{(u)}$, we need to emphasize that in our scheme $B^{(u)}$ not only records the identity (ID_i) of the replication node considered by u but also contains the number of shared starting keys with ID_i.

2) Detection of The Node Replication Attack (EDD-On-line-Step): Each node broadcasts its ID_i periodically to execute the EDD-On-line-Step. Each node uses a timer t locally to record the elapsed time in each time interval . The initiator

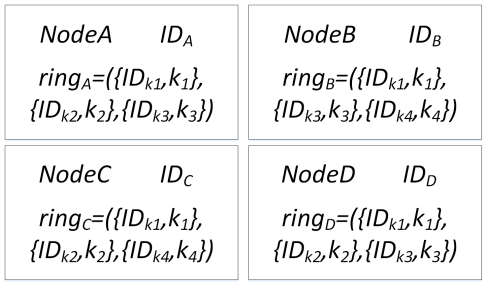


Fig. 2. Information of ring in nodes.

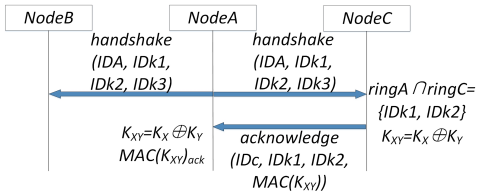


Fig. 3. A-B-C handshake.

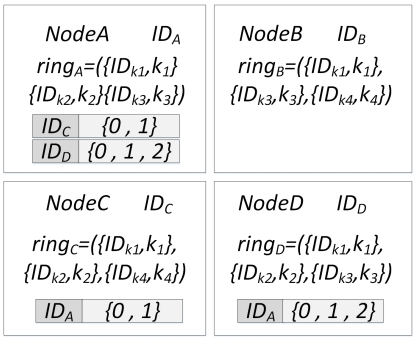


Fig. 4. Pairwise key establishment.

time is t_0. For the time interval T, if $t > T$, then $t = t_0$, which means the beginning of a new interval. The algorithm of EDD is shown as Algorithm 3.

The term $B^{(u)}$ is stored in the message sent to the base station to further contend with the node replication attack.

Algorithm 3 : EDD-On-Line-Step

broadcast beacon b_u
if $t \neq t_0$ **then**
  receive beacons b_{v_1},b_{v_d}
  while $1 < \kappa < d$ **do**
    $\mathcal{L}^{(\mu)}[\upsilon_\kappa] = \mathcal{L}^{(\mu)}[\upsilon_\kappa] + 1$
  end while
  if $\mathcal{L}^{(\mu)}[\upsilon_\kappa] > \psi$ **then**
    set $\mathcal{B}^{(\mu)} = \mathcal{B}^{(\mu)} \cup \{\upsilon_\kappa\}$
  end if
else
  set $\mathcal{L}^{(\mu)}[s_\kappa] = 0$, $\kappa = 1 \ldots n$
end if

3) Voting: Each node broadcasts a check message to neighboring nodes, and neighboring nodes send an acknowledge message as a return. The node that has received an acknowledge message calculates the number of neighboring nodes n'. The node puts n' and the number of starting keys into the ring r. Then, it puts $B^{(u)}$ into the message m and sends m to the base station for voting.

The base station calculates $d(d_i)$ of each node (ID_i) by $pro = \frac{d}{n'}$, $n = \frac{r}{pro}$, and $n = \frac{(rn')}{d}$ and updates the value of d. Let the notation of $T_{vote} = min\{\psi, min(d_i)\}$ be the voting threshold. Then, the base station counts all $B^{(u)}$ received and updates the count of voting of each node. For ID_i, if the $count > T_{vote}$, the base station will hold ID_i as a replication node. Then, execute the work of revocation.

E. The Revocation Phase

In this phase, the base station broadcasts the ID_i of the replication nodes and the ID_{k_i} of keys of the ring in the replicas. The neighboring nodes of the replicas will break off the link with the replicas and forbid the shared keys with replication nodes. The ID_{k_i} of the ring in replication nodes will be forbidden by the base station . Moreover, these ID_{k_i} in the pool are also forbidden.

Moreover, the recovery of system is very important. The base station will forbid too many keys as the number of replicated nodes increases, which decreases the number of keys available in the key pool. This situation will affect the connectivity of the sensor system. We need to eliminate those malicious replicate nodes to prevent the reduction in the number of available keys from affecting the connectivity of node systems. Since the nodes are physically deployed, we need to manually remove the malicious nodes. After those malicious replicas are removed, the operator notifies the base station, which in turn restores the disabled keys.

Then, SRKD updates the value of s. SRKD counts all shared keys in all the replication nodes with ID_i in every $B^{(u_i)}$ and calculates the max value. The max value will be the new value of s and will be used in the next deployment to prevent the further node replication attack.

F. The Prevention Phase

To mitigate the subsequent damage of the node replication attack to the sensor system, we need to establish a prevention mechanism against the node replication attack. We use the key characteristics of communication between nodes to do that. In the next deployment, if the number of shared keys is greater than or equal to the new s, then our SRKD will not establish the link between the two nodes and the base station will revoke the malicious replicas. Because every node picks the ring randomly from the key pool, it is less likely for two nodes having more than s common keys. Thus, we believe that the node replication attack occurs. This process can relieve the harm that replicas may inflict on the network to a certain extent.

V. Performance Analysis

A. Resilience

In this part, we evaluate the resilience about the ability to oppose the presence of compromised secret information.

The following formula defines the probability that the opponent can eavesdrop on the link between nodes uncompromised.

QC link [15]:

$$\begin{aligned}&[\sum_{i=q}^{r}\binom{r}{i}\binom{p-r}{r-i}\binom{p}{r}^{-x}\\&\sum_{j=0}^{i}\binom{i}{j}(-1)^j\binom{p-j}{r}^{x}]\\&/[\binom{p}{r}-\sum_{j=0}^{q-1}\binom{r}{j}\binom{p-r}{r-j}]\end{aligned} \tag{1}$$

QSC link [16]:

$$\begin{aligned}&[\sum_{i=q}^{r}\binom{r}{i}\binom{p-r}{r-i}\binom{p}{r}^{-x}\\&\sum_{j=0}^{\mathrm{Min}(i,s)}\binom{Min(i,s)}{j}(-1)^j\binom{p-j}{r}^{x}]\\&/[\sum_{k=q}^{r}\binom{r}{k}\binom{p-r}{r-k}]\end{aligned} \tag{2}$$

Our SRKD link:

$$[\sum_{i=q}^{s}\begin{pmatrix} r \\ i \end{pmatrix}\begin{pmatrix} p-r \\ r-i \end{pmatrix}\begin{pmatrix} p \\ r \end{pmatrix}^{-x} \sum_{j=0}^{\mathrm{Min}(i,s)}\begin{pmatrix} Min\,(i,s) \\ j \end{pmatrix}(-1)^j\begin{pmatrix} p-j \\ r \end{pmatrix}^{x}] /[\sum_{k=q}^{r}\begin{pmatrix} r \\ k \end{pmatrix}\begin{pmatrix} p-r \\ r-k \end{pmatrix}] \quad (3)$$

The formula below defines the probability of a forged node passing the authenticity confirmation in QC, QSC and SRKD.

$$1-\sum_{j=1}^{q} j\begin{pmatrix} r \\ q=j \end{pmatrix}(-1)^j\begin{pmatrix} p-r+q-j \\ r \end{pmatrix}^{x}/\begin{pmatrix} p \\ r \end{pmatrix}^{x} \quad (4)$$

Similar to the scheme of QSC, the resilience against the eavesdropping of the secret information provided by our proposed SRKD can be regarded as a general case of that provided by QC.

The formula is composed of the following:

(1) There is the whole $Min\,(i\ ,\ s)$ starting keys used by the link in the rings of x nodes compromised. This situation is calculated as follows:

1) The formula of $\begin{pmatrix} p \\ r \end{pmatrix}^{-x}\sum_{j=0}^{Min(i,s)}\begin{pmatrix} Min\,(i,s) \\ j \end{pmatrix}(-1)^j\begin{pmatrix} p-j \\ r \end{pmatrix}^{x}$ equals 1 when $j=0$;

2) It should subtract the probability that at least one key for the link that is not included in the rings of x compromised nodes, and this probability is related to the iterations of the summation. Additionally, the redundant items at the subsequent iterations of the summation are corrected by subtracting or adding $\left((-1)^j\right)$ combinations alternatively.

(2) Multiply the probability that two nodes that shared i keys can be eavesdropped. The pairwise key consists of $Min\,(i,s)$ starting keys because $q<i<s$ in our proposed scheme. The probability is related to the following two cases:

1) $\left(\begin{pmatrix} r \\ i \end{pmatrix}\begin{pmatrix} p-r \\ r-i \end{pmatrix}\right)$ is the number of combinations of i shared starting keys that can establish the pairwise key that is between two rings of two nodes on the link.

2) It should divide $\left(\sum_{k=q}^{s}\begin{pmatrix} r \\ k \end{pmatrix}\begin{pmatrix} p-r \\ r-k \end{pmatrix}\right)$ which is the entire amount of possible combinations of k shared starting keys between the two nodes on the link.

The probability of a forged node passing the authenticity confirmation shown by the formula is the same for SRKD and QSC. For both schemes, an opponent that shares at least q starting keys and establishes a link with a node may pass the authenticity confirmation. The calculation of the formula of SRKD is that one minus the probability that in the node that would perform the authenticity confirmation, there are less than q starting keys in the ring shared with the x compromised nodes. It is calculated as follows:

(1) Calculate the number of cases that at most $q-j$ keys are shared in every iteration.

1) On the first phase, at most $q-1$ keys are shared:

(a) The formula $\begin{pmatrix} r \\ q-j \end{pmatrix}$ is the number of combinations composed by q-1 keys;

(b) It is multiplied by the number of sets that are in the entire amount of x rings in the node that only include keys in a set of $q-1$ keys or out of the node that will perform combination.

2) After the first phase of iteration, less than $q-1$ keys are shared as the other cases. In addition, the redundant items will be corrected in the subsequent phases:

(a) The redundant items at the subsequent iterations of the summation $\sum_{j=1}^{q} j\begin{pmatrix} r \\ q=j \end{pmatrix}(-1)^j\begin{pmatrix} p-r+q-j \\ r \end{pmatrix}^{x}$ are corrected by subtracting or adding $\left((-1)^j\right)$ combinations alternatively;

(b) Inside a ring $\begin{pmatrix} r \\ q=j \end{pmatrix}$, it is equal to the number of combinations of q-j keys;

(c) Multiply the number of sets in which the entire amount of x rings that only include keys in a set of $q-j$ keys, or out of the node that would perform the confirmation;

(d) Multiply the number of items in which every set has been considered (j) redundantly.

(2) Divide the all possible sets of starting keys that the opponent $\begin{pmatrix} p \\ r \end{pmatrix}^{x}$ has owned.

The probability of passing the authenticity confirmation is equal to QC and QSC if $q=1$.

For the resistance of the node replication attack, Yu et al. [12] has demonstrated the security and practicability of EDD. Moreover, considering the network latency, packet losses and other environment factors, we improve the resistance method by introducing a voting mechanism to make a second confirmation that can increase the accuracy of examination. The value $min\,\{\psi, min\,(d_i)\}$ is the threshold in the voting mechanism, and $min\,(d_i)$ means the minimum expected number of links that a node can establish in the key-setup phase. For some nodes in the network, the situation of $d_i<\psi$ may occur, which makes the replicas hard to be detected in EDD. Therefore, we use $min\,\{\psi, min\,(d_i)\}$ as the voting threshold to respond to the situation of $d_i<\psi$. In this situation, we can make a judgment of malicious replicas if nodes acquire more than d_i votes.

Then, we use the "Security Protocol Animator for Automated Validation of Internet Security Protocols and Applications" (SPAN+AVISPA)(SA) [22] and a role-oriented language, high level protocol specification language (HLPSL) to analyze the security of the key distribution phase of SRKD.

The process of validating SRKD using SA is shown in Algorithm 4. For more details, please refer to [21].

The result is shown in Fig. 8. The proposed SRKD is simulated by the OFMC backend, which is a great verification paradigm that can analyze, for example, the Dolev-Yao model and replay attack. The indication in the analysis result demonstrates that our scheme is "SAFE".

B. Detection and Revocation Evaluation of Malicious Replication Nodes

We simulate our scheme by using OMNet++. In our simulation, there are 50, 100, 150, and 200 nodes in our simulated systems. Then, we inject 20 replicas for each system to test the effectiveness of our defense method. The results demonstrate

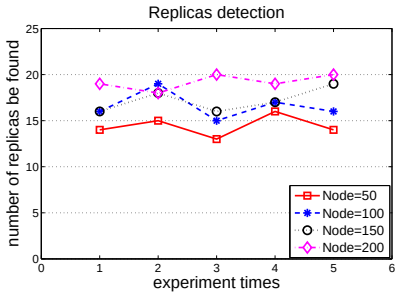


Fig. 5. Experimental result of replicas detection.

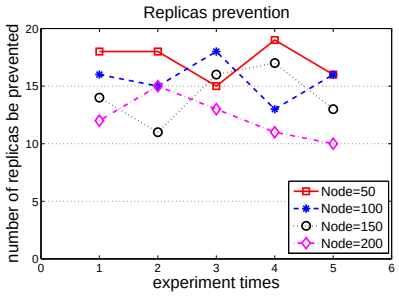


Fig. 6. Experimental result of replicas prevention.

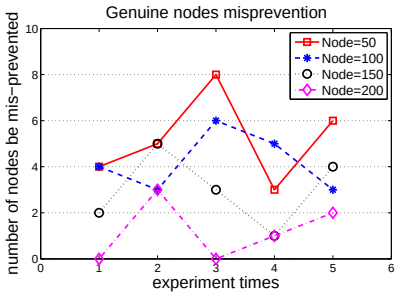


Fig. 7. Experimental result of genuine nodes misprevention.

that our proposed scheme can resist node replication attacks to a very large extent.

SUMMARY	STATISTICS
SAFE	parseTime: 0.00s
BACKEND	searchTime: 0.43s
OFMC	visitedNodes: 91 nodes
COMMENTS	depth: 10 plies

Fig. 8. Experimental result under OFMC backend.

Our scheme can resist node replication attacks from three aspects: detection, revocation and prevention of replication nodes. We evaluate the detection and revocation against malicious replication nodes in this section.

Algorithm 4 : Verifying the SRKD in SPAN+AVISPA

1. Simplify the communication in the system to the communication between two nodes
2. PART 1: Role Definition: NodeA/B
3. : agents in the system, parameters stored in NodeA/B, encryption key, hash function, transmission channel
4. : local network environment, initial state of network
5. : list of network state transitions
6. PART 2: Role Definition: Session
7. : agents in the system, parameters stored in agents, encryption key, hash function
8. : local network environment, transmission channel
9. : the agents composition of the session
10. PART 3: Role Definition: Environment
11. : local network environment, transmission channel, parameters stored in agents in the system environment, encryption key, hash function, security parameters
12. : parameters stored in agents in the system environment, intruder knowledge
13. : the sessions composition of the environment
14. PART 4: Role Definition: Goal
15. : security parameters
16. PART 5: Run Environment

Fig. 5 and Table I show the number of replicas detected in the networks with different numbers of nodes. It can be seen that as the ratio of normal nodes increases, the accuracy of detection increases. We also find that if there is a small number of nodes in the network, the threshold ψ of EDD may be greater than $min\,(d_i)$. Here, the node with $d = min\,(d_i)$ at the edge of the network may not establish at least $min\,(d_i)$ links, which means a replica cannot obtain more than $min\,(d_i)$ votes. As the number of nodes in the network increases, it reduces the probability of occurrence of nodes ($d = min\,(d_i)$). As a result, the probability that the replicas are deployed near those nodes ($d = min\,(d_i)$) decreases.

TABLE I. The Detection Result of SRKD

Num of nodes	50	100	150	200
Num of replicas	20	20	20	20
Num of replicas detected (Expt.1)	14	16	16	19
Num of replicas detected (Expt.2)	15	19	18	18
Num of replicas detected (Expt.3)	13	15	16	20
Num of replicas detected (Expt.4)	16	17	19	19
Num of replicas detected (Expt.5)	14	16	19	20
Normal node ratio	71.43%	83.33%	88.24%	90.91%
Avg detection ratio	72%	83%	86%	96%

The detection rate of the existing state-of-the-art detection schemes are shown in Table II (N denotes the number of nodes). It can be seen that the detection rate of our SRKD is at a medium level when the number of nodes in a network is less than 100. However, SRKD has a good performance when the nodes are greater than 200. Although TDD and SDD appear to have a perfect result, it is only shown in some particular cases. TDD and SDD also suffer from the efficiency limitation. Most importantly, our SRKD scheme is based on random key distribution, and it is easy to integrate with other random key distribution schemes.

Regarding the revocation, we assume that the replication nodes can be successfully revoked once they are detected.

TABLE II. The Comparison of Detection Rates of Different Schemes

Scheme	N< 100	N> 200	Scheme	N< 100	N> 200
RM [17]	63%	95%	P-MPC [18]	86%	89%
LSM [17]	55%	90%	TDD [20]	-	100% particular case
RED [19]	87%	90%	SDD [20]	-	100% particular case
LINE [18]	62%	74%	EDD [12]	70%	95%
SDC [18]	70%	95%	SRKD	72%	96%

C. Prevention Evaluation of Malicious Replication Nodes

The reason that we use the $max\,(s)$ in the prevention part is to guarantee the normal deployment of genuine nodes by reducing the accuracy of detection appropriately. Fig. 6 and Fig. 7 illustrate the number of replicas prevented and the number of genuine nodes misprevented, respectively. If the network has more nodes, the threshold of $max\,(s)$ can cover a larger field and can better represent the generality of all the nodes in the network. Therefore, it can reduce the number of mispreventions. However, as the $max\,(s)$ increases, some replicas that have not shared more than $max\,(s)$ starting keys cannot be prevented. This part needs to be improved in the future.

D. Other Evaluation

1) Storage Efficiency Evaluation: We compare the storage efficiency of SRKD with EG, QC and QSC and further evaluate the efficiency of SRKD in this section. The following parameters are used in the evaluation.

- l_k: the length of the key,
- l_{kID}: the length of the identifier of the key,
- l_{ID}: the length of the identifier of the node,
- l_T: the length of the time interval T,
- l_ψ: the length of threshold ψ,
- l_{num}: the length of counter,
- r: the number of keys in ring,
- $maxrg$: the maximum number of the starting keys for the node to establish a link,
- $maxrp$: the maximum number of the replicate nodes in the neighbour field.

The memory costs are shown in Table III. In the prestorage of EG, the storage of r keys is the main part of the storage required. In the prestorage of QC, both the starting keys and the pairwise keys need to be stored in the memory. QC also stores a hash function. However, it does not store any identifier of the pairwise key because the neighbor nodes in the future will use the shared keys to establish a new link. In the working storage part, the difference between EG and QC are the storage of l_k l_{kID}. Moreover, l_k requires more memory than l_{kID}. In the prestorage of QSC, it stores the starting keys, those identifiers, the identifier of the node and the key for updating. In the working storage of QSC, the number of identifiers of those keys used to establish the pairwise key is denoted as maximum ($maxrg$), which can provide a pessimistic examination for QSC.

In SRKD, the prestorage is the same as QSC, but the working storage is different. In SRKD, $v*l_{num}$ is the memory requirement of $L^{(u)}$ and $maxrp\,(l_{ID}+l_{num})$ is the memory requirement of $B^{(u)}$. Along the same reasoning as $maxrg$, SRKD considers $maxrp$ as the number of the neighbor nodes that have been considered as replicas. The number of identifiers can be lower than $maxrg$ and the number of replicas can be lower than $maxrp$.

TABLE III. Comparison in Memory Cost

Protocol	Pre-storage	Working storage
EG [14]	$r*(l_k+l_{kID})$	$r*(l_k+l_{kID})+$ $v*(l_{ID}+l_{kID})$
QC [15]	$r*(l_k+l_{kID})$	$r*(l_k+l_{kID})+$ $v*(l_{ID}+l_k)$
QSC [16]	$r*(l_k+l_{kID})+$ l_k+l_{ID}	$r*(l_k+l_{kID})+v*$ $(l_{ID}+maxrg*l_{kID})+$ l_k+l_{ID}
SRKD	$r*(l_k+l_{kID})+$ l_k+l_{ID}	$r*(l_k+l_{kID})+v*$ $(l_{ID}+maxrg*l_{kID})$ $+l_k+l_{ID}+l_T+l_\psi+$ $v*l_{num}+$ $maxrp\,(l_{ID}+l_{num})$

For the convenience of a comparison, we consider the following case used in QSC: $r=10$, $v=10$, $maxrg=5$, $maxrp=5$, $l_k=16\ bytes$, $l_{ID}=2\ bytes$, $l_{kID}=1\ byte$, $l_T=1\ byte$, $l_\psi=1\ byte$ and $l_{num}=1\ byte$. The comparison is shown in Fig. 9.

EG is the first random key distribution, and the storage overhead is low; however, the security is insufficient. QC makes up for the lack of safety, but it consumes too much storage. QSC greatly optimizes the storage efficiency on the basis of QC. We can find that SRKD requires much lower storage than QC from Fig. 9. Although it requires slightly higher storage than QSC, the increase in storage is within acceptable limits for security enhancements (against a replication attack).

2) Communication Efficiency Evaluation: In this part, we evaluate the communication overhead of SRKD. Let l_b denote the length of the beacons that is required to be exchanged in EDD and l_σ denote the length of the signature $\sigma=(R,y,z)$.

In the scheme of EG, QC, QSC and SRKD, it requires two one-hop transmissions to establish a link. In SRKD, the handshake message stores the identifier of the node l_{ID} and the identifiers of keys in the ring $r*l_{kID}$. The acknowledge message stores the identifier of the sender of handshake message l_{ID}, the MAC of message l_k and the identifiers of the selected keys $maxrg*l_{kID}$ that are used to establish the link. We use the $maxrg$ to make a pessimistic examination.

The communication efficiency for different schemes is shown in Table IV. In SRKD, there are two kinds of communication, one is "node to node" and the other one is "node to base station". We consider $r=10$, $maxrg=5$, $l_k=16\ bytes$, $l_{ID}=2\ bytes$, $l_{kID}=1\ byte$, $l_\sigma=41\ bytes$ and $l_m=17\ bytes$. In our message recovery scheme, the point on the elliptic curve is defined over the finite field F_q, where $|q|=20.5\ bytes$ (For ECC, a general safety requirement is approximately 200-bit). From the affine coordinate, the elliptic curve point is also represented as $Q=(x,y)$. In SRKD, the signature $\sigma=(R,y,z)$ consists of two values of the coordinate in Z_q and an elliptic curve point. However, R is preloaded into the sensor node during the Extract phase. Therefore, the length of $\sigma=(R,y,z)$ l_σ can be calculated as follows: $|\sigma|=|y|+|z|=20.5\ bytes+20.5\ bytes=41\ bytes$. The message that the node sends to the base station consists $B^{(u)}$ and the identifier of the node. Therefore, the formula is $l_m=maxrp*(l_{ID}+l_{num})+l_{ID}$.

TABLE IV. Communication Comparison

Scheme	Transmission size
EG [14]	$(r+1)*l_{kID}+2*l_{ID}+l_k$
QC [15]	$2r*l_{kID}+2*l_{ID}+l_k$
QSC [16]	$(r+maxrg)*l_{kID}+2*l_{ID}+l_k$
SRKD	$(r+maxrg)*l_{kID}+2*l_{ID}+l_k+l_b$

TABLE V. Communication Cost Between Node and Base Station

Method	Transmission size
With message recovery	$l_\sigma+l_{ID}+maxrg*l_{kID}$
Without message recovery	$l_\sigma+l_m$

We can find that SRKD has a lower communication overhead than QC from Fig. 10. For the explanation, please refer to the discussion on the storage efficiency evaluation. As shown in Fig. 11 and Table V, we can obviously find that the message recovery scheme has reduced the cost of communication.

VI. Industrial Application

We consider the monitoring and controlling of temperature in an oil refinery as an example. It is important to monitor the temperature of the workshop and oil can, since petrochemicals are characterized by combustion and explosion components.

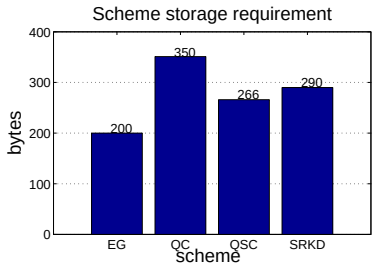


Fig. 9. Storage cost comparisonOur proposed scheme SRKD, EG [14], QC [15] and QSC [16].

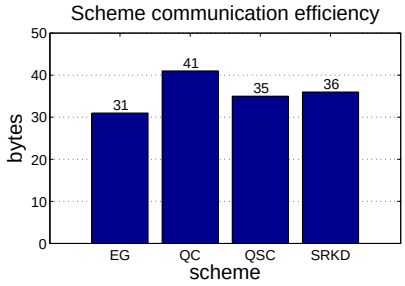


Fig. 10. Communication comparison: Our proposed scheme SRKD, EG [14], QC [15] and QSC [16].

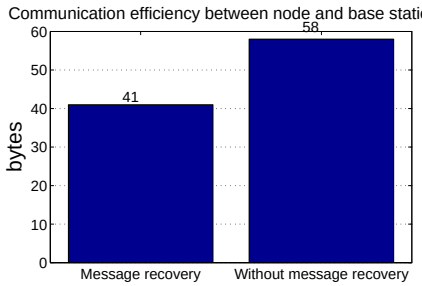


Fig. 11. Communication cost between node and base station.

There are many temperature sensors used to detect the temperature in this context. These sensors collect temperature data and send them to a control center that will regulate the temperature by following some safety threshold. However, as shown in Fig. 12, if the sensor network is exposed to the node replication attack, replicas may replace the high temperature data with the low one within the threshold and further send the false information to the control center to make the center think that everything is under control. Since the control center does not regulate the high temperature in time, the excessively high temperature may cause destruction to facilities and even endanger the local economy and people's lives.

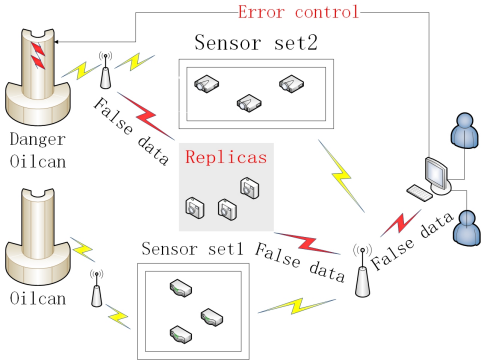


Fig. 12. Node replication attack in oil refinery WSS.

Our SRKD scheme can be used as a solution to effectively prevent industrial applications from replication attacks. Specifically, it runs the random key distribution and prepares the related parameters used in the defense mechanism against the node replication attack. SRKD can detect the attack in a timely manner and revoke the replicas as soon as possible after the deployment of nodes. The iterative updating of parameter s can be used to prevent new node replication attacks in the next deployment to a certain extent. Moreover, the smaller storage and communication requirement in our SRKD scheme may help industries reduce costs in upgrading hardware facilities.

VII. Conclusion And Future Work

We have proposed a novel random key distribution scheme called SRKD that has a higher level of resilience compared to other random key distribution schemes w.r.t. node capture and information eavesdropping. There are two outstanding advantages of the proposed SRKD. (1) It provides a defense mechanism against the node replication attack. SRKD can effectively provide the detection and revocation of replicas, and at the same, it can prevent replication nodes from injecting "false information" to a certain extent. (2) SRKD requires low energy consumption. Both the storage and communication overhead in SRKD are lower than those of the classical QC, although the cost of storage and communication is slightly higher than that of QSP. Nevertheless, we state that this is the trade-off: increasing an insignificant cost to achieve a higher level of security. Moreover, we use the message recovery mechanism to reduce the bandwidth cost yielded by SRKD. In practice, SRKD may help us strengthen the security of WSS applications without significantly jeopardizing key bootstrapping and efficiency. In future work, we need to improve several aspects. For example, we consider efforts aimed at selecting a better voting threshold for replication detection, optimizing the upper bound s to ensure the connectivity of network, and maximizing the prevention of the attack.

References

[1] G. Xu, Y. Cao, Y. Ren, X. Li, and Z. Feng, "Network security situation awareness based on semantic ontology and user-defined rules for internet of things," *IEEE Access*, vol. 5, pp. 21 046–21 056, 2017.

[2] H. Radhappa, L. Pan, J. Xi Zheng, and S. Wen, "Practical overview of security issues in wireless sensor network applications," *International Journal of Computers and Applications*, vol. 40, no. 4, pp. 202–213, 2018.

[3] L. Pan, X. Zheng, H. Chen, T. Luan, H. Bootwala, and L. Batten, "Cyber security attacks to modern vehicular systems," *Journal of Information Security and Applications*, vol. 36, pp. 90–100, 2017.

[4] X. Cao, J. Chen, Y. Xiao, and Y. Sun, "Building-environment control with wireless sensor and actuator networks: Centralized versus distributed," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3596–3605, 2010.

[5] X. Zeng, G. Xu, Z. Xi, X. Yang, and W. Zhou, "E-aua: An efficient anonymous user authentication protocol for mobile iot," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2018.

[6] G. Xu, Y. Zhang, A. Sangaiah, X. Li, A. Castiglione, and X. Zheng, "Csp-e2: An abuse-free contract signing protocol with low-storage ttp for energy-efficient electronic transaction ecosystems," *Information Sciences*, vol. 476, pp. 505–515, 2019.

[7] X. Liu, N. Xiong, N. Zhang, A. Liu, H. Shen, and C. Huang, "A trust with abstract information verified routing scheme for cyber-physical network," *IEEE Access*, vol. 6, pp. 3882–3898, 2018.

[8] G. Xu, L. Jia, Y. Lu, X. Zeng, Z. Yao, and X. Li, "A novel efficient maka protocol with desynchronization for anonymous roaming service in global mobility networks," *Journal of Network & Computer Applications*, vol. 107, p. S1084804518300407, 2018.

[9] J. Ren, Y. Zhang, K. Zhang, A. Liu, J. Chen, and X. Shen, "Lifetime and energy hole evolution analysis in data-gathering wireless sensor networks," *IEEE transactions on industrial informatics*, vol. 12, no. 2, pp. 788–800, 2016.

[10] J. Luo, J. Hu, D. Wu, and R. Li, "Opportunistic routing algorithm for relay node selection in wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 112–121, 2015.

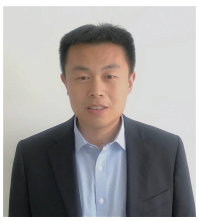
[11] L. Oliveira, J. Rodrigues, A. deSousa, and V. Denisov, "Network admission control solution for 6lowpan networks based on symmetric key mechanisms," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2186–2195, 2016.

[12] C. Yu, Y. Tsou, C. Lu, and S. Kuo, "Localized algorithms for detection of node replication attacks in mobile sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 5, pp. 754–768, 2013.

[13] K. A. Shim, "Basis: A practical multi-user broadcast authentication scheme in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1545–1554, 2017.
[14] L. Eschenauer and V. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*. ACM, 2002, pp. 41–47.
[15] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *IEEE Security and Privacy*. IEEE, 2003, pp. 197–213.
[16] F. Gandino, R. Ferrero, and M. Rebaudengo, "A key distribution scheme for mobile wireless sensor networks: q-s-composite," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 34–47, 2017.
[17] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in *IEEE Security and Privacy*. IEEE, 2005, pp. 49–63.
[18] B. Zhu, S. Setia, S. Jajodia, S. Roy, and L. Wang, "Localized multicast: Efficient and distributed replica detection in large-scale sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 7, pp. 913–926, 2010.
[19] M. Conti, R. DiPietro, L. Mancini, and A. Mei, "A randomized, efficient, and distributed protocol for the detection of node replication attacks in wireless sensor networks," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2007, pp. 80–89.
[20] K. Xing and X. Cheng, "From time domain to space domain: Detecting replica attacks in mobile ad hoc networks," in *IEEE INFOCOM*. IEEE, 2010, pp. 1–9.
[21] ashdawn, "Srkd," https://github.com/ashdawn/SRKD, accessed December, 2018.
[22] AVISPA, "Automated validation of internet security protocols and applications," http://www.avispa-project.org/, accessed December, 2018.

Longpeng Li is a master's student at the College of Intelligence and Computing, Tianjin University, China. He received his B.S. degree from the School of Information Science and Technology, Yanshan University of China in 2017. His current research are random key distribution scheme in WSN and cryptography.

Guangquan Xu is a Ph.D. and full professor at the Tianjin Key Laboratory of Advanced Networking (TANK), College of Intelligence and Computing, Tianjin University, China. He received his Ph.D. degree from Tianjin University in March 2008. He is a member of the CCF and IEEE. His research interests include cyber security and trust management.



Litao Jiao received his MBA degree in 2016 from Shandong University of Science and Technology. He is now an associate professor in Qingdao Huanghai University, China. He has been awarded the prize of Provincial Educational Achievement in year 2018, participated in 5 major provincial and municipal research projects, and published more than 10 papers. His research interests include: HR management and information security.

Xiaotong Li is a masters student at the College of Intelligence and Computing, Tianjin University, China. She received her B.S. degree from the School of Mechanical, Electrical and Information Engineering, Shandong University of China in 2018. Her current research interests include web application protection technique.

Hao Wang is an associate professor in Norwegian University of Science and Technology, Norway. He has a Ph.D. degree and a B.Eng. degree, both in computer science and engineering. His research interests include big data analytics, industrial internet of things, high performance computing, safety-critical systems, and communication security. He has published 80+ papers in reputable international journals and conferences. He served as a TPC co-chair for IEEE DataCom 2015, IEEE CIT 2017, ES 2017 and reviewers for journals such as IEEE TKDE, TII, TBD, TETC, T-IFS, IoTJ, and ACM TOMM. He is a member of IEEE IES Technical Committee on Industrial Informatics. His webpage is www.haowang.no.

Jing Hu received the Ph.D. degree from the School of Computer Science and Technology, Tianjin University, in 2013. She is a lecture of the College of Intelligence and Computing, Tianjin University. Her main research directions include formal method, IoT, and software engineer.

Hequn Xian received Ph.D degree in the Institute of Software, Chinese Academy of Sciences in 2009. He was a visiting scholar with college of information science and technology, the Pennsylvania State University. His research interests include cryptography, cloud computing security, and network security.

Wenjuan Lian is a Ph.D. and associate professor at the College of Computer Science and Engineering, Shandong University of Science and Technology. She received her masters degree (2002) and doctors degree (2011) from Shandong University of Science and Technology. Her research focuses on deep learning, cyber security, etc. She has published 20+ papers in core journals and international conferences, finished 10+ national and provincial projects, published three books.

Honghao Gao received the Ph.D. degree in Computer Science and started his academic career at Shanghai University in 2012. He is an IET Fellow, BCS Fellow, EAI Fellow, IEEE Senior Member, CCF Senior Member, and CAAI Senior Member. Prof. Gao is currently a Distinguished Professor with the Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education, China. His research interests include service computing, model checking-based software verification, and sensors data application.