Research Article

The Novel Authentication Scheme Based on Theory of Quadratic Residues for Wireless Sensor Networks

Chen-Xu Liu, Yun Liu, Zhen-Jiang Zhang, and Zi-Yao Cheng

Key Laboratory of Communication & Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Chen-Xu Liu; hbtslcx@gmail.com

Received 2 August 2012; Revised 3 March 2013; Accepted 3 March 2013

Academic Editor: Lei Zhang

Copyright © 2013 Chen-Xu Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Being one of the main technologies for Internet of Things (IoT), wireless sensor network is absolutely critical and plays an increasingly important role in the development of Internet of Things. Wireless sensor network is mobile network composed of sensor nodes with limited resources, like computation capability, energy, memory storage, and communication capability. Its characteristic results in that the security of wireless sensor network faces a great challenge. Hence, the security of wireless sensor network is an important issue for the development of IoT and shall be focused on. In this paper, a novel authentication scheme is proposed for wireless sensor networks based on theory of quadratic residues. It is making the use of master key to achieve simple symmetric cryptographic primitives and authentication operation and reach the aims of great resistance against the attacks and low energy consumption. Then the proposed scheme is compared with other network-wide key management schemes, obtaining better results in the aspects of security, efficiency, and scalability.

1. Introduction

The future Internet, designed as “Internet of Things (IoT),” is foreseen to be a wide world network of interconnected objects which is uniquely addressable in various types of communication networks [1]. Internet of Things will lead to great influence and opportunities to change and improve people’s lives and pushes the technologies in various areas developing rapidly. Playing an important role in the future developments of IoT, wireless sensor network can collect the information in the surrounding environments and process and transmit the data to the Internet. It is an important data resource for IoT and the development of IoT cannot be lack of wireless sensor network’s support. Such a technology endows the WSN a new perspective and challenge. Therefore, wireless sensor network is more and more popular and important in technologies and applications of IoT. However, wireless sensor network is composed of small, lightweight, inexpensive sensor nodes. In the system, those sensor nodes have limited resources, like low computation capability, limited energy, limited memory storage, and narrow bandwidth [2, 3]. Meanwhile, it is usually deployed in untrusted and unsecure areas where communication is needed but there are no steady communication infrastructures or those infrastructures are absent. Therefore, we not only pay our attention to making wireless sensor networks useful, manageable, efficient, and feasible, but also focus on the improvement of security.

Security plays an important role in the wireless sensor networks’ technologies [4–8], because sensor nodes are deployed mostly in hostile and tactical scenarios. Due to the nature of wireless sensor networks, they must face lots of great challenges about the security, such as limited resource of sensors, wireless transmission nature, and low resistance to attacks, which will result in that adversaries can easily eavesdrop the data transmission, impersonate other users, inject bogus data, and alter contents of legitimate messages. So the administrators must adopt effective measures to keep the security of WSNs, such as data confidentiality, data integrity, privacy, and authentication.

Therefore, authentication mechanisms need to be implemented to protect sensor nodes from various malicious attacks. While the WSN is running, when a user wants to join the WSN, it needs to be first authorized by the nodes in the WSN so that accessing the network illegally does
not happen. Therefore, in recent years many researchers pay more attention to key management and authentication for WSN and make good achievements in the research field. A lot of novel, safe, and efficient authentication protocols are proposed in recent years, such as C. T. Li, M. S. Hwang and Y. P. Chu’s scheme [9, 10], Das’s schemes [11], and Dressler’s scheme [12].

1.1. Threat Model. Due to the nature of wireless sensor network, it is vulnerable to resource consumption attacks and usually not resistant to those attacks, which facilitates physical manipulation and key material stealing. Few of the attacks are of the most concern for WSN [13] and are listed as follows.

(1) **Physic Attack.** Sensor nodes are usually deployed in hostile environment and exposed easily to physical intrusion with the adversary. Physic attack is gaining access to nodes’ memory to extract the master key of legitimate sensor nodes in the network [14, 15].

(2) **Node Capture Attack.** In a WSN, an attacker can capture sensor nodes and extract desired information from nodes’ memories or obtain access to the network. Once a set of sensor nodes is under the control of the attacker, this attacker has the capability of eavesdropping private information of the nodes and recovering cryptographic and authentication key [16–18].

(3) **Denial-of-Service Attack.** In a WSN, a denial-of-service attack (DoS attack) is an attempt to make a sensor node unavailable to its intended users [19–22]. The attackers will try to send external authentication requests to sensor nodes and want them to respond to those requests, such that it cannot respond to legitimate requests or responds so slowly as to be rendered effectively unavailable or use up the sensor nodes’ energy and make authenticators cannot work normally. It prevents the sensor nodes from being authenticated by a legitimate authenticator and makes the authentication scheme not work.

(4) **Reply Attack.** An attacker can eavesdrop on the conversation between the claimant and the authenticator and retransmit the message to the legitimate authenticator as being authentic. It leads to legitimate access for the attackers into the network and no response to the claimant so that the authentication scheme cannot work normally.

(5) **Impersonation Attack.** Attackers can masquerade as legal nodes to be authenticated or authenticate others by intercepting and falsifying data and thereby gain illegal advantages.

1.2. **Our Contributions.** This paper aims to propose a novel authentication scheme based on quadratic residues for wireless sensor networks, which is utilizing the master key to calculate the authentication key chains. Our proposed scheme is to achieve simple key management and to provide an authentication operation for the user access to the network. It utilizes the master key to generate pairwise keys in initialization and key setup phase and then erases the master key from the memory and does not need master key during the normal operation of the network anymore. In addition, proposed authentication model has a good resistance to those attacks which are of the most concern for WSN. Therefore, a security analysis has been given to demonstrate that proposed scheme can withstand the previously mentioned attacks and have good scalability. Meanwhile, we make a simple summary and analysis of energy consumption and find it have better performance compared with other schemes.

1.3. **Organization of Our Paper.** The rest of the paper is organized as follows. In Section 2, we reviews and describes exiting related works. In Section 3, an improved authentication scheme is proposed based on quadratic residues. Security analysis and performance evaluation are provided in Section 4. Our conclusions are presented in Section 5.

2. **Related Works**

This section introduces two types of previously published work: master-key-based key management protocols and authentication schemes based on quadratic residue. Those protocols make important contributions in this area and have been discussed widely.

2.1. **Master-Key-Based Key Management Protocols.** SPINS is a suite of security protocols optimized for wireless sensor networks [8, 23]. It is presented to achieve data confidentiality, two-party data authentication, evidence of data freshness and authenticated broadcast for severely resource-constrained environments. In SPINS, it is assumed that there is a trusted third party in the network and each node shares its master key with this trusted third party. Thus, the trusted third party has the capability of achieving the authentication and generating pairwise keys for nodes. When two nodes want to establish a pairwise key to make a communication with each other, they must ask trusted third party to guarantee the legality of each other. BROSK is an energy efficient master-key-based key management protocol [24]. In the protocol each node establishes a pairwise key with its neighbors by broadcasting key negotiation messages instead of being authenticated through the trusted third party. Once one node receives a key negotiation message, it can construct the shared pairwise key with who broadcasts this message by generating the MAC. Both protocols are proposed based on master key however, they do not take the security of master key into consideration, because it is vulnerable to physical attacks in two protocols. Once the attackers extract the master keys, they can obtain all the information of key management to compromise the whole network. Zhu et al. [25] proposed the Localized Encryption and Authentication Protocol (LEAP) which utilizes four types of keys for each sensor node. These are used for different purposes and ranges. (1) Individual key is shared between each node and base station. (2) Pairwise key is shared between a node and its neighbor node. (3) Group key is shared by all nodes in the network. (4) Cluster key is a key between a node and its all neighbor nodes. Among four types of keys, pairwise key is more important and its security guarantees the security of LEAP. However,
Kim et al. [26] pointed out that in LEAP, the entire network will suffer a severe loss if the initial master key is exposed to an attacker during key setup. So they propose a quick key establishment to improve the security of LEAP. Our proposed scheme will avoid this drawback and be perfectly resistant to those attacks.

2.2. Authentication Scheme Based on Quadratic Residue. Recently Chen et al.’s scheme [27] is proposed to achieve mutual authentication based on hash function and quadratic residues assumption for RFID system. The scheme is utilizing direct indexing for each tag’s authentication and avoiding servers’ brute search. In condition, Chen et al.’s scheme has the capability of having good resistance to those attacks and resolution to security problems, like TID anonymity, individual location privacy, replay attack, mutual authentication and DOS attack. Soon after Chen et al.’s scheme is proposed, Yeh et al. [28] demonstrate the weaknesses of that scheme in their work and present an improved scheme to avoid those already existing problems. According to the description of Yeh et al.’s work, Chen et al.’s authentication scheme is vulnerable to impersonation attack. In condition, Chen et al.’s scheme cannot effectively resist location privacy and replay attack. So Yeh et al.’s scheme makes an additional supplement of Chen et al.’s scheme and utilizes the number generated by the tags to add in the session between the tags and the server. The solution obtains a good effect in the authentication scheme and makes it invulnerable to impersonation attack.

Two schemes described previously are proposed based on quadratic residue however, they are both specific to RFID system and not suitable for WSN system. RFID system does not take the security of master key into consideration. And the common attacks in two systems are different so that we need to improve the authentication scheme invulnerable to those common attacks in WSN system. On the other hand, the relationship between the server and the tags in RFID system is one-to-many; in other words, the tags must be authenticated by one server. However, that is different in WSN system and each legal and authenticated sensor node should have the capability of authenticating the new nodes who want to join the network. Our proposed scheme will consider those drawbacks in detail and solve those problems.

3. Proposed Authentication Scheme

In the system, it is assumed that there is not a trusted server and each sensor node directly negotiates a session key with neighboring nodes and uses the master key to authenticate itself with its neighbors. And there are three phases in this paper: key predistribution phase, network deployment phase, and authentication phase. While the authentication protocol is operating, there are authentication messages exchanged among the nodes to be used for pairwise key generation and new node authentication. That message packets format used within the network is shown in Figure 1. The meanings of those acronyms in the Figure 1 are as follows: SRC, source address; DST, destination address; LEN, message length; MT, message type; AM, authentication message. When a node broadcasts its randomly selected number or begins the authentication operation, it will send the message packet to other neighboring nodes.

On the clarity and conveniences of this paper, the representations are listed in the following.

(i) \(k_M\) is the network-wide master key.

(ii) \(h(M)\) is the hash function of message \(M\) and \(h_K(M)\) is the hash function of message \(M\) with key \(K\).

(iii) \(h(M)\) is represented as that message \(M\) is hashed \(i\) time without key \(K\).

(iv) \(h_K(M)\) is represented as that message \(M\) is hashed \(i\) time with key \(K\).

(v) \(k^{(i)}_{auth}\) is the authentication key of \(j\)th cycle authentication.

(vi) \(C^j_i\) is \(i\)th tuple of \(j\)th cycle authenticator.

Then a detailed description for those phases is shown in the following.

3.1. Key Predistribution Phase. Key predistribution phase is carried out before the nodes deployed in the network, because a network-wide symmetric master key \(k_M\) is preloaded and stored in each node and is needed to generate a session key for the authentication. In proposed scheme, a mechanism is defined as the authenticators that authenticate the nodes and make the nodes access the network legally. In fact, the authenticator is essentially the especial counter, which records the number and progress of those authenticated nodes. Each legal node in the network has its own authenticator. It is calculated with the value of tuple and cycle in the node. Each authenticator is composed of a set of numbers, including \(l\) tuples of random numbers in current cycle and the results of hash function with authentication key over those numbers. Once a new node has been authenticated successfully by a legal node, the value of tuple in this legal node is incremented and this tuple of current cycle authenticator is closed and never used. If the authentication fails, the authenticator in this tuple of current cycle remains open. When a node runs out of \(n\) tuples of current cycle in this authenticator, it shall generate a new set of \(l\) tuples in the next cycle authenticator set.

The symbol \(C^j_i\) is expressed as \(i\)th tuple of \(j\)th cycle authenticator. The authenticator is initialized in each node as \(0\)th cycle authenticator before network deployment and denoted by \(C^0_i\). Master key is assigned to the value of authentication key in \(0\)th cycle authenticator. Authentication key and authenticator set in \(0\)th cycle are calculated:

\[
k^0_{auth} = h^0(k_M) = k_M,
\]

\[
C^0_i = \left\{ (r_j, h^0_{auth}(r_j)) \right\} = \left\{ (r_j, h_{0_M}(r_j)) \right\}, \quad i = 0, 1, \ldots, l - 1,
\]

(1)
where \( r_i \) is one of \( l \) tuples of random numbers in current cycle and 0th cycle authenticator is composed of \( r_i \) and hash function of \( r_i \) with authentication key \( k_M^{\text{auth}} \).

The authentication key is changed with the cycle increasing and the authenticator set in current cycle is removed after new authenticator set in next cycle is generated by the node.

### 3.2. Authenticator Update

In this section, the process to update authentication key and authenticator set for \( i \)th tuple of \( j \)th cycle is introduced in the following.

1. Update current authentication key to a new one with hash functions, getting
   \[
   k_{\text{auth}}^{j+1} = h(k_{\text{auth}}^j) = h^j(k_M) = h^j+1(k_M).
   \]  
   (2)

2. Calculate new authenticator for \( i \)th tuple of next cycle with newly updated authentication key \( k_{\text{auth}}^{j+1} \):
   \[
   C_i^{j+1} = \{(r_i, h_{\text{auth}}^{j+1}(r_i))\}, \quad i = 0, 1, \ldots, l - 1.
   \]  
   (3)

3. Obtain the new key:
   \[
   k_{\text{auth}}^{j+1} = h^{j+1}(k_M), C_i^{j+1} = \{(r_i, h_{\text{auth}}^{j+1}(r_i))\}
   \]  
   \[i = 0, 1, \ldots, l - 1.\]  
   (4)

### 3.3. Network Deployment Phase

The nodes in the network are deployed randomly in the environment during the beginning of this phase. The network structure of WSN is shown in Figure 2. The nodes in the WSN are divided into various clusters. Then the nodes will find their neighboring nodes in their cluster and exchange session messages with neighboring nodes to initialize the network after the deployment. The details are listed in the following.

1. Update session key: each node updates its session key from 0th cycle to 1st cycle of authenticator and generates the session key. Session key for each node is including authentication key and authenticator for current cycle. The session key shall be updated as authentication key in first cycle authenticator after the deployment. For example, session key of node \( A \) in 0th tuple is \([k_M, C_0 = (r_i, h_{\text{auth}}(r_i))]\) in initialization phase, and then it is replaced by \([k_{\text{auth}}^1 = h(k_M), C_1 = (r_i, h_{\text{auth}}^1(r_i))]\).

2. Generate pairwise keys: after authentication key is updated, each node randomly selects a random number to generate pairwise keys with neighboring nodes based on master key. The generation is shown in Figure 3. In Figure 3, node A has 8 neighboring nodes and broadcasts the message \( M_A \) to those neighboring nodes. Each node broadcasts its own random number with neighboring nodes for a short period of time. Setting the reasonable period of time is to keep the attackers who listen to random number broadcasting from overhearing and intercepting in this step [29]. For example, node A randomly selects a number \( r_A \) and broadcasts the number to the neighboring node \( B \). Broadcasted message \( M_A \) is \( ID_A[r_A]||ID_A[r_A]K_M \), including node A’s ID, the random number \( r_A \), and the encryption values of ID \( A \) and \( r_A \) with master key \( k_M \). Node \( B \) also broadcasts its message \( M_B = ID_B[r_B]||ID_B[r_B]K_M \) to neighboring nodes. When node \( A \) receives the node \( B \)’s message, node \( A \) can generate the pairwise key \( K_{AB} = [r_A]r_BK_M \) with node \( B \).

So each node utilizes received random number to generate a list of pairwise keys of its neighboring nodes. After pairwise keys list is established, the master key is erased from memory and replaced by authentication key. Authentication key is changing with the time of hash function changing, which could eliminate the risk of storing master key.

3. After the deployment of the network, each node stores a series of keys, including its own encryption key, a list of pairwise keys of its neighboring nodes, its authentication key, and its session key in first cycle authenticator.

Following the previously steps, each node could begin to communicate with its neighboring nodes using pairwise key. In addition, each node is capable of authenticating new node.
that tries to join the network and confirming whether new node is a legal and permitted user.

3.4. Authentication Protocol Phase. During the authentication operation, node A is assumed as a new node that wants to join the network and stores network-wide master key, its 0th cycle authenticator. Node B is selected as a decision maker to confirm if node A is legal user. Node A must pass the authentication of node B, while node A shall also judge node B not to be a compromised and hostile user. The whole process is a two-way authentication and they must authenticate each other.

(1) Node A randomly selects a number \( r_A \) and sends a message \( M_1 \) including this random number to node B:

\[
M_1 = r_A. \tag{5}
\]

(2) Node B receives the message \( M_1 \) from node A and makes the following calculation and operation.

**Step 1.** Generate a pairwise key \( k_{AB} \) shared with node A.

**Step 2.** Select a random number \( t \) and choose \( i \)th tuple random number \( r_i \), and obtain authentication key \( k_{auth}^i \) and \( j \)th cycle authenticator:

\[
C_j^i = \left\{ (r_j, h_{auth}^j(r_j)) \right\}. \tag{6}
\]

**Step 3.** Calculate the values of \( x = h(k_{auth}^j) \oplus r_A \oplus r_i \oplus t \) and \( y = r_i \oplus t \) with current authentication key in 0th cycle.

**Step 4.** Compute the quadratic residue of \( x, r_i, t \) and to hide those numbers to guarantee the security not to be revealed during the transmission: \( X = x^2 \ mod \ n, R = (r_i^2 \ mod \ n)\oplus t \), and \( T = t^2 \ mod \ n \).

**Step 5.** Send the message \( M_2 \) encrypted by pairwise key \( k_{AB} \) to node A, including \( X, R, T, \) hash function of \( x, y, t \) with authentication key \( k_{auth}^j \) and the current cycle of authenticator, \( j \):

\[
M_2 = h_{auth}^j(x), h_{auth}^j(y), h_{auth}^j(t), j, X, R, T. \tag{7}
\]

(3) Node A receives the message \( M_2 \) from node B and performs the following operations.

**Step 1.** Update current cycle authenticator with this message and compute authentication key from current cycle to \( j \)th cycle with hash function:

\[
k_{auth}^j = h(k_{M2}). \tag{8}
\]

**Step 2.** Calculate \( x \) and \( t \) based on the parameters \( X \) and \( T \) in message \( M_2 \), obtaining the group \( \{x_1, x_2, x_3, x_4\} \) and \( \{r_1, r_2, r_3, r_4\} \). Then node A compares \( h_{auth}^j(x_p), h_{auth}^j(t_p) \) with \( h_{auth}^j(x) \) and \( h_{auth}^j(t) \) in \( M_2 \), where \( p = 1, 2, 3, 4 \). Finally, node A obtains the values of \( x \) and \( t \).

**Step 3.** Determines the value of \( r_i \) with the values of \( R, t, \) and \( y \). Firstly, obtain all the possible values of \( r_i \), like the group \( \{r_{i1}, r_{i2}, r_{i3}, r_{i4}\} \). Next, compare \( h_{auth}^j(r_{ip} \oplus t) \) with \( h_{auth}^j(y) = h(r_i \oplus t) \), where \( p = 1, 2, 3, 4 \), and confirm the value of \( r_i \).

**Step 4.** Authenticate whether node B is a legal user. At first, calculate the authentication key of node B: \( k_{auth}^j = x \oplus r_A \oplus r_i \oplus t \). Then compare \( k_{auth}^j \) with \( k_{auth}^j \) and make the following decision for node B. (1) If \( k_{auth}^i \not= k_{auth}^j \), node A will consider node B to be a comprised node and could not authenticate node B successfully. Node A will terminate this authentication operation with node B and wait for a certain time period to start another authentication process with other nodes. (2) If \( k_{auth}^i = k_{auth}^j \), go to next step to calculate and send response message to node B.

**Step 5.** Generate a response message, showing \( x_{response} = k_{auth}^i \oplus r_i \oplus t \). Then response message \( M_3 \) is sent to node B:

\[
M_3 = x_{response}. \tag{9}
\]

(4) Node B receives message \( M_3 \) and judges whether node A is a legal node. Calculate \( x' = k_{auth}^i \oplus r_i \oplus t \), compare \( x_{response} \) with \( x' \), and make the authentication. If \( x_{response} \not= x' \), node A should not be authenticated unsuccessfully by node B. Otherwise, node A is verified by node B as a legal user. Then update the values of tuple \( i \) and cycle \( j \). If all \( l \) tuples authenticators in \( j \)th cycle run out, the value of cycle \( j \) shall be incremented to \( j+1 \); that is, a new cycle authenticator is starting. Therefore, there are two possible conditions listed in the following.

If all \( l \) tuples of random number in \( j \)th cycle authenticator run out, update \( j \) and authentication key \( k_{auth} \) to \( j+1 \) and \( k_{auth} \). The \( j \)th authenticator \( C_j^i = \{ (r_i, h_{auth}^j(r_i)) \} \) is replaced by \( C_j^{i+1} = \{ (r_i, h_{auth}^j(r_i)) \} \). If there also remain other tuples of random numbers in \( j \)th cycle authenticator, \( i \)th tuple of random number \( r_i \) is updated to another tuple of random number \( r_{r_2} \), where \( e \in \{0, 1, \ldots, l - 1\} \). The authenticator \( C_j^i = \{ (r_i, h_{auth}^j(r_i)) \} \) is replaced by \( C_j^i = \{ (r_{r_2}, h_{auth}^j(r_{r_2})) \} \).

Then the whole authentication operation is shown in Figure 4.

4. Security Analysis and Performance Evaluation

In this section we make the security analysis and performance evaluation of our scheme and compare it with other schemes, like SPINS [8, 23], BROSK [24, 30], LEAP [25], KMS [26], and LWAS [31]. The reason why we choose those schemes is that all the schemes are based on network-wide key. The comparisons of various security attributes among our scheme and other works are listed in Table 1, while the performance comparisons are revealed clearly in Table 2.

4.1. Security Analysis. As we all know for this area, there are a certain number of attacks [14, 15] and security issues to be ubiquitous and inevitable in wireless sensor networks. Then our proposed authentication scheme will be proved to be resistant to those most common attacks in Section 4.1.

Secrecy. Due to good protection for transmitted data, the resident data \( (r_i, j, k_{auth}^j, C_j^i) \) in the nodes could not be
Retrieved during the communication. Only the legal nodes could get the related data $x$, $r_i$, $t$ by using the theory of quadratic residues. Even if the attackers get the data $x$, $r_i$, $t$, $k_{\text{auth}}$, they could not obtain the master keys of those nodes because of one-way property of the hash function. And $x$, $r_i$, $t$, $k_{\text{auth}}$ will be modified with beginning next round of the authentication operation and the data retrieved by those attackers could not make sense for next authentication. Hence, secrecy for our authentication scheme is guaranteed.

**Forward Secrecy.** The data transmitted between applicant and decision maker is well protected so as to be obtained difficulty by the adversary from the transmission. To prove forward secrecy of our authentication scheme, we assume that the attacker compromises the decision maker and obtains the data in this decision maker. Let the decision maker's current data be $(r_0, j, k_{\text{auth}}^1, C_0)$. The value of $r_0$ is selected randomly for each cycle by decision-maker and we could not use current value of $r_i$ to derive the value in previous cycle so that the attacker is not able to track the previous authentication. According to the value of $k_{\text{auth}}^1$, the value of $k_{\text{auth}}^1$ is not exposed to the attacker, because hash function is one way only and the values of the hash are unique for the hashed message. Meanwhile, the master key $k_M$ is erased after the deployment phase and attackers could not use the value of $j$ to obtain authentication key of each cycle. Therefore, attackers will not be able to find the previous authentication data using the current data $(r_0, j, k_{\text{auth}}^1, C_0)$.

**Resistance to Node Capture Attack.** Sensor nodes in most applications are mostly deployed in public or hostile environment. Most nodes are low cost and not tamper resistant. Therefore, the adversary may have the capability of taking physical control of nodes undetectably and compromise the cryptographic keys. As a result, this type of attack [16–18, 30] is considered as the main threat and the resistance against node capture is regarded as an important criterion to protect the security of the network. The performance of the resistance toward node capture is evaluated in the work [30] by H. Chan, A. Perrig, and D. Song. They calculate the fraction of total network communications that are compromised by a capture of $x$ nodes not including the communications in which the compromised nodes are directly involved. By utilizing this method, we could get the performance of our scheme's resistance against node capture. This metric shows our scheme's resistance against this attack is pretty good, because those nodes erase their master keys before the authentications begin and then the capture of nodes could not obtain any information, like master keys and authentication keys of next cycle, about the links that are directly involved.

**Resistance to Physical Attacks.** In proposed scheme, nodes’ master keys are erased after the deployment of the network. So there are no master keys in those nodes before the authentication phase. Sensor nodes deployed in public or hostile environment must be exposed easily to physical intrusion with the adversary. However, due to the erased master keys, those nodes are vulnerable to physical attacks [14, 15] and the information in them is unrelated to direct linked nodes.

**Resistance to Replay Attack.** In proposed scheme, both the decision-maker and applicant generate a random number during the authentication. The random number generated by node $B$ is added to the session tokens $x_{\text{response}}$ responded by node $A$. The session token of each authentication processing

---

**Table 1: Security comparisons.**

| Scheme | Node authentication | Node revocation | Node capture attack | Secrecy | Forward secrecy | Replay attack | DoS attack | Physical attack | Impersonation attack | Overall security |
|--------|---------------------|-----------------|---------------------|---------|----------------|---------------|-----------|-----------------|---------------------|----------------|
| SPINS  | Yes                 | Very difficult: revocation of master key | No                   | Yes     | No             | Yes           | Yes       | No              | No                   | Low             |
| BROSKE| Yes                 | Very difficult: revocation of master key | No                   | Yes     | No             | Yes           | Yes       | No              | No                   | Low             |
| LEAP   | Yes                 | Easy: delete the key            | No                   | Yes     | No             | Yes           | Yes       | No              | No                   | Medium          |
| KMS    | Yes                 | Easy: delete the key            | Yes                  | Yes     | Yes            | Yes           | Yes       | Yes             | No                   | Medium          |
| IWAS   | Yes                 | Easy: delete the key            | Yes                  | Yes     | Yes            | Yes           | Yes       | Yes             | No                   | High            |
| Ours   | Yes                 | Easy: delete the key            | Yes                  | Yes     | Yes            | Yes           | Yes       | Yes             | Yes                  | High            |

**Table 2: Security comparisons.**

| Scheme | Scalability | Computation cost | Communication overhead | Energy consumption |
|--------|-------------|------------------|------------------------|--------------------|
| SPINS  | No          | PRF              | $n \cdot (2L_{ID} + 2L_{\text{NONCE}} + 3L_{\text{MAC}})$ | High               |
| BROSKE| Yes         | $H + PRF$        | $L_{ID} + L_{\text{NONCE}} + L_{\text{MAC}}$ | Low                |
| LEAP   | Yes         | $2H + 2PRF$     | $L_{ID} + L_{\text{NONCE}} + L_{\text{MAC}}$ | Low                |
| KMS    | Yes         | $2H + PRF$      | $L_{ID} + L_{\text{NONCE}} + L_{\text{MAC}}$ | Medium             |
| IWAS   | Yes         | $H + PRF$       | $L_{ID} + L_{\text{NONCE}}$ | Low                |
| Ours   | Yes         | $H + PRF$       | $L_{ID} + L_{\text{NONCE}}$ | Low                |
Applicant node A

\[ r_A, j, P, q, n, k_i^j, C_i \]

Decision maker node B

\[ r_B, j, n, k_i^j, C_i \]

1. randomly chooses \( t \) and randomly selects \( r_A \)

\[ (r_A, h(k_A^{j+1}(r_A))) \]

2. \( x = h(k_A^{j+1}(r_A)) \oplus r_A \oplus r_t \oplus t \)

3. \( y = r_t \oplus t \)

4. \( X = x^2 \mod n \)

5. \( T = t^2 \mod n \)

6. \( R = (r_t^2 \mod n) \oplus t \)

7. Computes \( k_A^{j+1}(x) = h(k_A^{j+1}(r_A)) \oplus r_A \oplus r_t \oplus t \)

8. \( k_A^{j+1}(y) = h(k_A^{j+1}(r_A)) \oplus r_A \oplus r_t \oplus t \)

9. \( k_A^{j+1}(y) = h(k_A^{j+1}(r_A)) \oplus r_A \oplus r_t \oplus t \)

10. Computes \( k_A^{j+1}(y) = h(k_A^{j+1}(r_A)) \oplus r_A \oplus r_t \oplus t \)

11. \( r_t = r_t \oplus t \)

12. If \( r_t \neq k_A^{j+1}(y) \), go to next step.

13. If \( r_t = k_A^{j+1}(y) \), node B has not been successfully authenticated and the authentication is aborted.

14. Calculates \( x_{response} = k_A^{j+1}(y) \oplus r_A \oplus r_t \oplus t \)

15. Calculates \( k_A^{j+1}(y) = h(k_A^{j+1}(r_A)) \oplus r_A \oplus r_t \oplus t \)

16. If \( k_A^{j+1}(y) \neq k_A^{j+1}(y) \), node B has not been successfully authenticated.

17. If \( n \) tuples in the authenticator run out update \( j \) and \( k_A^{j+1}(y) \), with \( j + 1 \) and \( k_A^{j+1+1}(y) \) authenticator being updated to \( G_i^{j+1} = \{(r_0, h(k_A^{j+1+1}(r_0))\} \)

18. Else update \( r_t \) to \( r_t \), where \( e \in [0, L, \ldots, I - 1] \) authenticator is updated to \( G_i^{j+1} = \{(r_t, h(k_A^{j+1+1}(r_0))\} \)

19. Else A has not been successfully authenticated.

Figure 4: The authentication protocol.
and synchronizes its authenticator with node B. Meanwhile, only node A has the value of \( p \) and \( q \) and is able to obtain the correct \( x, r, t \) according to Chinese Remainder Theorem. So those guarantee node A to pass the node B’s authentication. To avoid attackers impersonating node A to pass the authentication of node B, node A sends a random number \( r_A \) to node B, while node B’s response is calculated with the random number \( r_A \). If the attackers impersonate node A to transmit another number to node B, node B will find node A’s response \( x_{\text{response}} \) in \( M_2 \) is wrong, because \( x_{\text{response}} \) is calculated with random number and master key and attackers do not have right master key. In addition, node A could judge whether the response in \( M_2 \) is right or wrong by using its master key so that node A could exclude those attackers to impersonate node B. Thus, proposed scheme will be of great resistance against impersonation attack.

After the analysis is discussed, the comparisons in the security of our scheme are summarized with others in Table 1. It is concluded in Table 1 that security levels of our scheme and LWAS are highest, followed by LEAP and KMS. SPINS and BROSK schemes are in lowest security in the comparisons. In addition, our scheme has great resistance against impersonation attack and is more secure than LWAS scheme.

### 4.2. Performance Evaluation

The performance evaluation for wireless sensor network is very important due to sensor nodes’ limited resource. So the analysis of energy consumption is key point in the performance evaluation for wireless sensor network.

As we all know, energy consumption of the security in wireless sensor network is mainly divided into two types: communication overhead and computation cost. However, it is concluded from previous and recent works that communication overhead plays a more important role in energy consumption than computation cost and accounts for a significantly larger part of energy consumption. For example, most of energy cost of adding SPINS protocol to wireless sensor network arises from communication and transmission overhead of extra data for security protocol rather than from computation cost. According to the work [8, 23], transmission cost is increased for SPINS protocol by 98%, while the MAC and encryption computation cost just only account for 2%. Thus, deceasing data transmission for security protocols and minimizing the communication overhead as soon as possible are one of the design goals for security protocols.

In this paper, the number of transmitted messages is utilized to make the evaluation of energy consumption, which is approximately equal to real energy consumption. We set the network in a certain area of 500 m × 500 m and deployed \( n \) sensors. Let us randomly select one node for analyzing the size of exchanged messages and estimating the communication overhead. At first, the lengths of different types of messages shall be defined: \( L_{\text{ID}}, L_{\text{NONCE}}, L_{\text{KEY}}, \) and \( L_{\text{MAC}} \) are the length of node ID, nonce, symmetric keys, and MAC, respectively. In addition, it is assumed that each node has \( n \) neighboring nodes in the network. Because the nodes exchange the messages with a trusted third party during the pairwise key establishment, the length of exchanged messages for those security protocols is significantly larger than others, which is expressed as \( n \cdot (2L_{\text{ID}} + 2L_{\text{NONCE}} + 3L_{\text{MAC}}) \). For BROSK, LEAP, and KMS protocols, node ID, nonce, and MAC need to be broadcasted to establish pairwise keys with neighboring nodes so that the size of transmitted messages is \( L_{\text{ID}} + L_{\text{NONCE}} + L_{\text{MAC}} \). Compared with the protocols previously, the establishment of pairwise keys just needs to broadcast node ID and nonce, so that the size of transmitted messages in LWAS and our proposal is \( L_{\text{ID}} + L_{\text{NONCE}} \). From the analysis of the size of transmitted messages, we can conclude that transmission cost of LWAS and our proposal is lowest in those protocols.

Besides analyzing communication overhead of those protocols, computation cost is also a part of estimating energy consumption, which is listed in Table 2. From the result in Table 2, it is concluded that computation cost of SPINS protocol is lowest, followed by BROSK, LWAS, our proposal, and others because the nodes in SPINS protocol are just only needed to generate a nonce and other computations are done by a trusted third party. Summarizing the previously analysis, our proposal’s energy consumption that is proportional to communication overhead is lower than others except LWAS protocol. In addition to lower energy consumption, guaranteeing the network scalable is also an important design goal. In SIPINS, pairwise keys are established by exchanging the messages with a trusted third party. Hence, the scale of the network in SPINS is not too large, or a lot of collisions will be generated in this network and the performance is degrading with the network running. Nodes in other protocols are only exchanging messages with neighboring nodes and are independent of network size. So proposed scheme scales very well with the size of the network and achieves good scalability for the network.

### 5. Conclusion

Being an important part of IoT, the security of wireless sensor networks should be focused on greatly. As wireless sensor networks are generally deployed in unsecured and untrusted areas, the transmitted data is vulnerable to all kinds of intrusions, like eavesdropping, interception, and modification. Those intrusions threaten the security and privacy of the users in the whole network and propose a great challenge for the development of wireless sensor network and IoT.

In this paper, our scheme is proposed to utilize the master key to achieve key management and the authentication operation based on quadratic residue for wireless sensor network, which is to provide the protection of the master key and offer the new users legal access to the network. It has a perfect resistance against those attacks which are of the most concern for WSN and achieves higher level of the security. Meanwhile, its energy consumption is in a lower position in comparison with other schemes. Therefore, it is concluded that our scheme achieves key management and node authentication, protects the node from a series of attacks.
which has reached a higher point in the security, and has very little increase in energy consumption.

Acknowledgments

This research is partly supported by National Natural Science Foundation of China under Grants 61071076, National High-Tech Research and Development Plans (863 Program) under Grants 2011A0401004-2, and the Academic Discipline and Postgraduate Education Project of Beijing Municipal Commission of Education.

References

[1] “Internet of Things in 2020: Roadmap for the Future,” 2008, http://www.smart-systems-integration.org/public/internet-of-things.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communications Magazine, vol. 40, no. 8, pp. 102–105, 2002.

[3] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” Computer Networks, vol. 52, no. 12, pp. 2292–2330, 2008.

[4] V. C. Giruka, M. Singhal, J. Royalty, and S. Varanasi, “Security in wireless sensor networks,” Wireless Communications and Mobile Computing, vol. 8, no. 1, pp. 1–24, 2008.

[5] A. Perrig, J. Stankovic, and D. Wagner, “Security in wireless sensor networks,” Communications of the ACM, vol. 47, no. 6, pp. 53–57, 2004.

[6] X. Du and H. H. Chen, “Security in wireless sensor networks,” IEEE Wireless Communications, vol. 15, no. 4, pp. 60–66, 2008.

[7] Y. Wang, G. Attebury, and B. Ramamurthy, “Securing wireless sensor networks: a survey,” IEEE Communications Surveys Tutorials, vol. 10, no. 3, pp. 2–23, 2006.

[8] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, “SPINS: security protocols for sensor networks,” Wireless Networks, vol. 8, no. 5, pp. 521–534, 2002.

[9] C. T. Li, M. S. Hwang, and Y. P. Chu, “An efficient sensor-to-sensor authenticated path-key establishment scheme for secure communications in wireless sensor networks,” International Journal of Innovative Computing, Information and Control, vol. 5, no. 8, pp. 2107–2124, 2009.

[10] C. T. Li, M. S. Hwang, and Y. P. Chu, “A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks,” Computer Communications, vol. 31, no. 12, pp. 2803–2814, 2008.

[11] A. K. Das, “Improving identity-based random key establishment scheme for large-scale hierarchical wireless sensor networks,” International Journal of Network Security, vol. 14, no. 1, pp. 1–21, 2012.

[12] F. Dressler, “Authenticated reliable and semi-reliable communication in wireless sensor networks,” International Journal of Network Security, vol. 7, no. 1, pp. 61–68, 2008.

[13] T. Kavitha and D. Sridharan, “Security vulnerabilities in wireless sensor networks: a survey,” Journal of Information Assurance and Security, vol. 5, no. 1, pp. 31–44, 2010.

[14] Z. Benenson, P. M. Cholewiński, and F. C. Freiling, “Vulnerabilities and attacks in wireless sensor networks,” in Wireless Sensors Networks Security, Cryptology & Information Security Series (CIS), pp. 22–43, IOS Press, Lansdale, Pa, USA, 2008.

[15] T. Kwon and S. H. Park, “Experimental study on wireless sensor network security,” in Intelligence and Security Informatics, vol. 3975 of Lecture Notes in Computer Science, pp. 741–743, 2006.

[16] P. Tague and R. Poovendran, “Modeling node capture attacks in wireless sensor networks,” in Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing, pp. 1221–1224, IEEE Press, September 2008.

[17] P. Tague and R. Poovendran, “Modeling adaptive node capture attacks in multi-hop wireless networks,” Ad Hoc Networks, vol. 5, no. 6, pp. 801–814, 2007.

[18] L. Eschenauer and V. D. Gligor, “A key-management scheme for distributed sensor networks,” in Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS ’02), pp. 41–47, ACM Press, November 2002.

[19] A. D. Wood and J. A. Stankovic, “Denial of service in sensor networks,” Computer, vol. 35, no. 10, pp. 54–62, 2002.

[20] D. R. Raymond and S. F. Midkiff, “Denial-of-service in wireless sensor networks: attacks and defenses,” IEEE Pervasive Computing, vol. 7, no. 1, pp. 74–81, 2008.

[21] J. Bellardo and S. Savage, “802.11 denial-of-service attacks: real vulnerabilities and practical solutions,” in Proceedings of USENIX Security Symposium, pp. 15–28, Washington, DC, USA, 2003.

[22] Q. Gu, P. Liu, and C. H. Chu, “Analysis of area-congestion-based DDoS attacks in ad hoc networks,” Ad Hoc Networks, vol. 5, no. 5, pp. 613–625, 2007.

[23] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, “SPINS: security protocols for sensor networks,” in Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom ’01), pp. 189–199, ACM Press, July 2001.

[24] B. C. C. Lai, D. D. Hwang, S. P. Kim, and I. Verbauwhede, “Reducing radio energy consumption of key management protocols for wireless sensor networks,” in Proceedings of the International Symposium on Lower Power Electronics and Design (ISLPED ’04), pp. 351–356, ACM Press, August 2004.

[25] S. Zhu, S. Setia, and S. Jajodia, “LEAP: efficient security mechanisms for large-scale distributed sensor networks,” in Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS ’03), pp. 62–72, October 2003.

[26] Y. H. Kim, H. Lee, D. H. Lee, and J. Lim, “A key management scheme for large scale distributed sensor networks,” in Proceedings of the IFIP TCS 6th International Conference (PWC ’06), vol. 4217 of Lecture Notes in Computer Science, pp. 437–446, Springer, 2006.

[27] Y. Chen, J. S. Chou, and H. M. Sun, “A novel mutual authentication scheme based on quadratic residues for RFID systems,” Computer Networks, vol. 52, no. 12, pp. 2373–2380, 2008.

[28] T. C. Yeh, C. H. Wu, and Y. M. Tseng, “Improvement of the RFID authentication scheme based on quadratic residues,” Computer Communications, vol. 34, no. 3, pp. 337–341, 2011.

[29] R. Anderson, H. Chan, and A. Perrig, “Key infection: smart trust for smart dust,” in Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP ’04), pp. 206–215, IEEE Press, October 2004.

[30] H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks,” in Proceedings of IEEE Symposium on Security and Privacy, pp. 197–213, IEEE Press, May 2003.

[31] O. Delgado-Mohatar, A. Fuster-Sabater, and J. M. Sierra, “A lightweight authentication scheme for wireless sensor networks,” Ad Hoc Networks, vol. 9, no. 5, pp. 727–735, 2011.
