Identity-based Key Management Scheme with Provable Security for Wireless Sensor Networks

Nanrun ZHOU*, Qiongxi JIANG, Xun CHEN

Department of Electronic Information Engineering, Nanchang University, Nanchang 330031, China

Abstract

Key management is the basis of security in wireless sensor networks. In the key management, more attentions should be paid to the public key schemes. An identity-based key management scheme for wireless sensor networks is proposed, where the node identity is used to encrypt the key generating material. The pairwise key is generated by the material ultimately. The security of the proposed scheme is analyzed with the provable security. It is proved that our scheme is IND-ID-CPA secure against some active attacks such as tampering and impersonation. The storage and communication overheads of our scheme are low enough to fit for wireless sensor networks. Addition and revocation of the nodes with backward-security and forward-security respectively make our scheme more feasible and flexible.

Keywords: Wireless Sensor Networks; Key Management; Provable Security; Identity-Based

1 Introduction

Wireless sensor networks (WSNs) have been implemented in battlefield, hospital, forest and other crucial fields. Various attacks with the principles in computer networks pose threats to WSNs [1, 2], moreover, there exist other attacks solely to damage the nodes in WSNs. Security is always the focus for WSNs, and key management with generation of the communication key is a main task of security. However, traditional security mechanism isn’t suitable for WSNs due to the limitation in energy, information process ability, and storage content of the node [3, 4, 5]. The communication key is mainly generated by the key predistribution scheme (KPS). The first feasible KPS for WSNs was proposed by Eschenauer and Gligor [6], where the communication key for any two nodes was computed by the shared keys stored in their memories. However, the probability of sharing keys between any two nodes is relatively low. The $q$-composite KPS presented by Chan and Perrig [7] makes the quantity of the shared keys amount for $q$ and improves the probability of sharing keys. The above-mentioned probability KPSs can’t guarantee any two nodes to generate the communication key. A combination-based deterministic KPS proposed by Camtepe [8] gave a solution to this problem. However, storage overheads of the node increase with the increment of key chain length as expanding the scale of WSNs [8].

*Corresponding author.
Email address: znr21@163.com (Nanrun ZHOU).

1548–7741/ Copyright © 2012 Binary Information Press
December 2011
Due to the unparalleled advantage of public key cryptography (PKC), key management schemes based on PKC can meet the maximum security requirement. Wander indicated that RSA cryptography and elliptic curve cryptography (ECC) can be used in the networks composed of low energy sensor nodes [9]. Liu realized an ECC-based key management scheme in the operation system TinyOS of the sensor node [10]. Du first mentioned the concept of c-neighbor that the sensor node just needs to generate the communication keys with two neighbor nodes routing to the base station [11]. One of the communication keys is the main key, and the other is the backup one. The overheads become much lower since the node needs not to generate the communication keys with all the neighbor nodes. Hu assembled the hardware TPM for the node to realize RSA cryptography in WSNs [12]. Rahman implemented the node verification employing ECC-based bilinear pairing computation [13]. These schemes in [9-13] indicate that PKC is reasonable for WSNs. Additionally, identity-based PKC can avoid to verify the public key with certification authority, which saves much energy for WSNs. Recently, Jia presented an identity-based encryption scheme for WSNs with provably security in standard model [14], where $q$-ABDHE problem was used as the computational complexity assumption. Moreover, the scheme was IND-sID-CPA secure. To save energy of nodes and enhance the security simultaneously, an identity-based key management scheme for WSNs is proposed in this paper. The bilinear Diffie-Hellman (BDH) problem is used as the computational complexity assumption and then our scheme is proved IND-ID-CPA secure.

There are two major contributions in this paper: (1) The existing cryptographic systems for WSNs concentrate on the two following aspects. The encryption schemes based on public key mechanisms from the traditional internet are introduced into WSNs. Meanwhile, the symmetric cryptographic schemes are studied all the time. In the public cryptographic systems, the nodes can not afford the sufficient energy and memory space so that the schemes cannot be well implemented. In the symmetric cryptographic ones, security is relatively lower. An identity-based key management scheme for WSNs is proposed to solve the aforementioned two problems. (2) Detailed theoretical studies on the security and performance of the proposed key management scheme for WSNs are given. It is shown that our scheme provides a general security.

This work is organized as follows. The basic knowledge is overviewed in Section 2, and our scheme is introduced in Section 3. In Section 4, the security of the proposed scheme is analyzed in detail, and the overheads are evaluated in Section 5. In Section 6, the mechanisms of revocation and addition are described. Finally, a brief conclusion is drawn in Section 7.

## 2 Preliminaries

Let $G_1$ and $G_2$ be the cyclic multiplicative groups of order $p$ which is a large prime. In addition, let $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear paring with the following properties [15, 16]:

1. Bilinearity: $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$, for $a, b \in \mathbb{Z}_q^*$ and $g_1, g_2 \in G_1$.
2. Non-degeneracy: There exists $g \in G_1$ so that $e(g, g) \neq 1 \in G_2$.
3. Computability: There exists an efficient algorithm to compute $e(g_1, g_2)$ for $g_1, g_2 \in G_1$.

**Definition 1** The bilinear Diffie-Hellman (BDH) problem is impossible to be solved during the polynomial time if $e(g, g)^{abc}$ can’t be computed given $(g, g^a, g^b, g^c)$ for $a, b, c \in \mathbb{Z}_q^*$. 
Definition 2 The scheme can achieve the IND-sID-CPA security if malicious nodes can’t distinguish the ciphertexts under the chosen plaintext attack with selectively chosen identity.

Definition 3 Compared with the IND-sID-CPA security, the scheme is called IND-ID-CPA secure if the ciphertexts are indistinguishable under the same attack with adaptively chosen identity.

The attack identities have been given for the adversary in the IND-sID-CPA security, while any identity can be chosen by the adversary as the attack objective in the IND-ID-CPA security. It is clear that the IND-ID-CPA security is relatively stronger.

Definition 4 A neighbor node $u$ is called the $c$-neighbor of another node $v$ if $u$ is a route from $v$ to the base station.

3 Identity-based Key Management Scheme for WSNs

In this section, an identity-based key management scheme for WSNs is presented. Our scheme is inspired by the key management schemes for WSNs [11, 14, 17] and the identity-based encryption schemes [18, 19, 20]. The proposed scheme includes five procedures: Setup, Extraction, Encryption, Decryption and KeyGen.

(1) Setup: The base station of WSNs generates public parameters for all the nodes. Assume that the number of nodes in WSNs is $N$. The base station generates $r_i, g_i^r$ ($i = 1, 2, \ldots, N$) and stores them in the memories of corresponding nodes, where $r_i \in R Z_q^*$. Then the base station chooses $a, b \in R Z_q^*$, where $h = g^a$, $k = g^b$. The public parameters $(g, h, k)$ are prewritten into all the nodes by the base station before the deployment of WSNs.

(2) Extraction: The base station generates the private keys for all the nodes in WSNs to secure the key generating materials. Two functions $H_1, H_2$ are chosen as $H_1 : \{0, 1\}^* \rightarrow G_1$, $H_2 : G_2 \rightarrow G_1$. Assume that there exists the universal node identity $u$, then the private key $d_u$ of $u$ can be computed as,

$$d_u = (d'_u, r_u) = ((H_1(u).k^{-r_u})^a, r_u).$$

Like the procedure in Setup, the base station also writes the private key into the corresponding node before the deployment of WSNs.

(3) Encryption: Assume that the node $v$ needs to build a communication key with its $c$-neighbor node $u$. $v$ chooses $c \in R Z_q^*$ and computes $t = g^c$. Then $v$ creates the ciphertext $C$ and transmits it to $u$.

$$C = (C_1, C_2, C_3) = (t, e(h, k)^c, g^r_v \oplus H_2(e(H_1(u), h)^c)).$$

(4) Decryption: $u$ decrypts the ciphertext $C$ by $d_u$ to get the key generating material $g^r_v$ with $v$.

$$g^r_v = C_3 \oplus H_2(e(d'_u, C_1), (C_2)^{r_u}).$$
(5) KeyGen: $u$ utilizes the random number $r_u$ stored in its memory to obtain the communication key $K_{uv}$, where

$$K_{uv} = (g^{r_v})^{r_u} = g^{r_u r_v}. \quad (4)$$

Similarly, $v$ gains $K_{vu}$ as

$$K_{vu} = (g^{r_u})^{r_v} = g^{r_u r_v}. \quad (5)$$

Apparenty, $K_{uv}$ is identical to $K_{vu}$. After generating the communication key, $u, v$ must delete $g^{r_v}$ and $g^{r_u}$, respectively.

The interaction between two neighbor nodes is shown in Fig. 1. Thus, the two nodes have generated the communication key of their own, and can use the key to encrypt/decrypt the message via the elementary symmetric cryptography.

4 Security Proof

The nodes of WSNs are usually deployed in the difficult circumstances, so there are different types of attacks, such as eavesdropping, tampering, impersonation, and the traditional man-in-the-middle attack etc. In the following, the attack model launched by the adversary is built including parameters generation, query for private keys, challenge and guess. The model is manipulated between the adversary $\alpha$ and the challenger $\beta$.

(1) Parameters generation. $\beta$ gets the public parameters $(g, h, k)$ and sends them to $\alpha$.

(2) Query for private keys in the first phase. After obtaining the public parameters, $\alpha$ chooses the node identity to be queried and sends it to $\beta$. Without loss of generality, we set the node identity as $u$. Once obtaining the request for query, $\beta$ chooses $x_1, y_1 \in Z_q^*$ demanding $H_1(u) = g^{x_1 k y_1}$, generates the private key $d_u = (h^{x_1}, y_1)$ of $u$ and sends it to $\alpha$.

$$h^{x_1} = (g^a)^{x_1} = (g^{x_1})^a = (H_1(u), k^{-y_1})^a. \quad (6)$$

(3) Challenge. After obtaining $d_u$, $\alpha$ chooses another node identity $v$ and two messages $m_0, m_1 \in_R G_1$, then sends them to $\beta$. On obtaining $v, m_0, m_1$, $\beta$ chooses $x_2, y_2 \in Z_q^*$ so that $H_1(v) = g^{x_2 k y_2}$ and computes

$$d_v = (h^{x_2}, y_2). \quad (7)$$

$\beta$ chooses $\lambda \in \{0, 1\}$ and sends $C_v$ to $\alpha$.

$$C_v = (t, Z, m_\lambda \oplus H_2(\alpha_2)). \quad (8)$$

$$\alpha_2 = e(h^{x_2}, t). Z y_2, Z \in G_2.$$
(4) Query for private keys in the second phase. After obtaining the ciphertext $C_v$, $\alpha$ queries the private keys of the nodes except $u, v$. $\beta$ returns the corresponding results of the queried nodes.

(5) Guess. To guess $\lambda$, $\alpha$ outputs $\lambda'$ to $\beta$ and verifies whether $\lambda'$ is identical to $\lambda$ or not. If $\lambda' = \lambda$, $\alpha$ wins the game launched by $\beta$.

The security of our scheme can be reduced to the BDH problem. Due to $h = g^a, k = g^b, t = g^c$, one can get

$$\alpha_2 = e(h^{x_2}, t).Z^{y_2} = e(H_1(v), h)^c. \quad (9)$$

under the condition $Z = e(g, g)^{abc}$. Therefore, the challenger can construct the reasonable ciphertext so that the scheme isn’t secure. Instead, $Z \in_R \mathbb{G}_2$, then $\alpha_2 = e(h^{x_2}, t).Z^{y_2}$ is random. In this case, the challenger can obtain the suitable ciphertext to guess $\lambda$ correctly with the maximum probability $q_h/p$ near to zero, where $q_h$ is the number of queries for $H_2$. As the BDH problem is apparently insolvable within the polynomial time, our proposed scheme is IND-ID-CPA secure.

5 Overheads Analysis

The overheads of the sensor node account for storage, communication and computation. Storage overhead is the necessary storage content for generating the communication key. In our scheme, public parameters, the random number, and the private key are stored in the node memory. Then storage overhead is a random number and five elements in group $\mathbb{G}_1$.

The computation overhead is the energy consumption for the communication key generation. Let $P_{bi}, P_e, P_*, P_\oplus$ be the overheads of bilinearity, exponentiation, scalar multiplication, xor operations, respectively. The computation overheads during the processes of encryption, decryption and the communication key generation are noted as $P_{En} = 3P_e + 2P_{bi} + P_\oplus + H_1 + H_2$, $P_{De} = P_{bi} + P_e + P_* + P_\oplus + H_2$ and $P_{KeyGen} = P_e$. Consequently, the total computation overheads is $5P_e + 3P_{bi} + 2P_\oplus + P_* + H_1 + 2H_2$.

The communication overhead can be defined as the energy consumption for transmitting and receiving the message. Let $e_{tr}$ and $e_{re}$ be the node’s energy needed to transmit and receive the message once. The communication overheads of our scheme is $e_{tr} + e_{re}$.

It indicates that the energy consumption in communication is much greater than that in computation in WSNs and the computation overheads can be saved by the optimally designed software. Thus, our scheme is suitable for WSNs since the general overheads of the node are low enough.

6 Revocation and Addition

The mechanisms of revocation and addition are indispensable in the key management for WSNs. Sensor nodes are subject to erosion by nature circumstance and become invalid in the open environment. Moreover the energy of common node is limited, such as 15kJ for the Mica node developed by the University of California, Berkeley [21]. The functions of revocation and addition are absolutely necessary in the security scheme for WSNs.
(1) Addition. Assume that $w$ is the node added. In the initialization, the base station generates $r_w, g^{r_w}$ and the private key $d_w$ of $w$.

$$d_w = (d'_w, r_w) = ((H_1(w).k-r_w)^a, r_w).$$  \hfill (10)

where $r_w \in R Z_q$. Then these parameters are written to the memory of $w$. After deployment, $w$ generates the communication key with the neighbor node routing to the base station as Fig. 1. Our scheme is backward-secure unquestionably since $w$ can’t get $r_u$ and $r_v$ to compute $K_{uv}$ even though obtaining $g^{r_u}$ and $g^{r_v}$.

(2) Revocation. Assume that $v$ is the node revoked. If $v$ is the $c$-neighbor of $u$, $u$ must seek for the backup node $v'$ and generate the communication key as well as delete $K_{uv}$. The method of key generation refers to Fig. 1. If $v$ is captured and therefore revoked by the base station, the adversary can only get $K_{uv}$, which is deleted by $u$. In addition, the adversary can’t attempt to impersonate the legitimate node and communicate with $u$ since the material $g^{r_u}$ has been deleted once generating $K_{uv}$. Hence, our scheme is forward-secure against the node capture.

7 Conclusion

Overheads, resistance and scalability are of concern during the generation process of the communication key for wireless sensor networks. In this paper, the identity-based key management scheme for WSNs is proposed. It just uses the node identity and random number produced by the base station to encrypt the key building material to generate the pairwise key. The security of our scheme is reduced to the BDH problem, thus, the scheme is provably secure against the active attacks. The overhead analysis shows that storage overhead and communication overhead required in our scheme are low. Specifically, the overheads are constant, which makes our scheme scalable.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (Grant No. 10647133), the Natural Science Foundation of Jiangxi Province (Grant No. 2009GQS0080), and the Research Foundation of the Education Department of Jiangxi Province (Grant No. GJJ11339).

References


