Key Distribution using Double Keyed-hash Chains for Wireless Sensor Networks

Jianmin Zhang¹, Jianwei Tan² and Jian Li¹
¹College of Computer, Henan Institute of Engineering, Zhengzhou 451191, China
²Department of Information Security, Henan Police College, Zhengzhou 451191, China
zjm1996@163.com, xzhn_2008@163.com, lj2006@163.com

Abstract

As a security primitive, key establishment plays the most crucial role in the designing of the security mechanism in the wireless sensor networks (WSNs). Unfortunately, the resource limitation of sensor nodes poses a great challenge for designing an efficient and effective key establishment scheme for WSNs. In this paper, we propose an effective key predistribution scheme using double keyed-hash chain schemes. In the proposed scheme, there are two key pools: upward key pools and downward key pools, and the key ring in each sensor is picked from these two key pools. The proposed scheme is analyzed based on connectivity, resistance against attacks, memory consumption and communication overhead. Comparing with the EG scheme and the q-composite scheme, our scheme performs better in terms of network resilience to node capture with little additional overhead cost.

Keywords: Wireless sensor networks; Key predistribution; Network security; Hash chain

1. Introduction

A wireless sensor networks (WSNs) is a collection of a large number sensor nodes that has limited computation, communication and power resources distributed in a designed area without any fixed structure [1-2]. Each sensor node is a small device that consists of data processing, sensing, and short range radio communication units, and a battery. Typically, sensor nodes are employed to collect environmental information. This information is aggregated by transferring it to other sensor nodes wirelessly until it reaches a sink node. Wireless sensor networks are candidates for many applications such as border security, military target tracking and scientific research in dangerous environment [3-5]. Since sensor nodes are often deployed in hostile locations, particularly with military applications, security is an essential issue in these networks. For instance, an adversary could capture sensor nodes, intercept transmitted messages, and propagate fake messages to the networks. Some researchers have studied the security issues of WSNs [6, 7]. Key management plays an important role on security in wireless sensor networks [8]. However, key management is WSNs is a difficult problem because of resource constraints of the sensor nodes.

In the literature, key management protocols are based on either symmetric or asymmetric cryptographic functions. Due to resource limitations in the sensor nodes, key management protocols based on public key cryptographic (asymmetric functions) are not appropriate [5]. A particular symmetric approach in WSNs is to use key predistribution with the sensor nodes, resulting in low cost key establishment. In this regard, various schemes have been proposed for key management in WSNs [7-17].
choice of a key management protocol should consider factors such as processing overhead, resource consumption and connectivity. However, some of these goals are contradictory [13, 18]. For example, increasing network connectivity also increases the memory requirements of sensor nodes. Hence, some techniques that provide high connectivity also consume significant memory and processing power. Others use less memory but have low connectivity.

In this paper, a new key management for WSNs is proposed using double keyed-hash chains. In this protocol, the upward key pool and downward key pool are generated and each sensor node is given two key rings: upward key ring and downward key ring. Upward keys provide upward secrecy since the adversary cannot learn upward keys in the lower class sensor nodes even if it compromise higher class sensor nodes. Similarly, downward keys provide downward secrecy since the adversary cannot learn downward keys in the higher class sensor nodes even if it compromise lower class sensor nodes.

The rest of this paper is organized as follows. The next section reviews the related works. Section 3 presents the development of our scheme. Section 4 describes the performances of our scheme, and Section 5 concludes the paper.

2. Selected Related Work

Eschenauer et al., [9] proposed the first random key pre-distribution scheme for WSNs, which we call EG scheme in this paper. In this scheme, during key predistribution scheme, a large key pool is initialized and some keys from a large key-pool are selected randomly and stored in the sensor nodes. After deployment, sensor may find a common key which can be used to establish secure communication. If not, the common key is established via an intermediate sensor node which has common keys with both of the sensor nodes. In this scheme, the capture of a node may lead to compromising a link between two no captured nodes, since these two nodes may have used the same key to secure their communication. To reduce the fraction of compromised links between noncompromised nodes, a modification to the random key scheme, denominated \( q \)-compromised scheme, is presented [10]. In this solution, for two nodes to establish a secure communication link, they are required to share at least \( q \) \((q>1)\) common keys in their key rings. The secret common key is the hash of the \( q \) common keys in their key rings.

Liu et al., proposed a new key predistribution scheme [11], which substantially improved the resilience of the network compared to the existing schemes. In this scheme, every sensor node is preloaded with coefficient of symmetric bivariate polynomial computed at one of its variables using its identification. The symmetry property of the polynomial allows two nodes to get their pairwise key respectively. This scheme exhibits a nice threshold property, which means that when the number of compromised nodes is less than the threshold, the probability that communications between any additional nodes are compromised is close to zero. A similar method was also developed by Du et al., [12], in which matrices are uses instead of polynomials. These two schemes are explored in [13-15].

Du et al., [16] developed a scheme using pre-deployment knowledge. In this scheme, they assume that the sensor nodes are deployed in groups of some sensor nodes over a rectangular area. In the key predistribution phase, the original key pool is divides into many smaller pools, each of which is associated to different group. These schemes can gain substantial improvement over exiting schemes that do not exploit deployment. This group-based deployment model is further deployed in [17, 18].
Another important work in the literature that aims to increase the network resilience without reducing secure connectivity was proposed in [19-22] for multiphase sensor networks. In multiphase sensor networks, WSNs are set up to function for longer period of time as compared to the lifetime of sensor nodes. So, new nodes need to be deployed in some intervals to provide continuity of network in multi-phase sensor networks.

3. The Proposed Scheme

In this section, we describe the proposed key predistribution scheme works in detail. The proposed scheme can be divided into three phases: key distribution phase, pairwise key establishment phase, and path key establishment phase. Although path-key establishment phase is the same as EG scheme and q-composite scheme, key predistribution phase and shared-key discovery phase are different in the previous schemes. The details of our scheme are described below.

3.1. Key Predistribution Phase

This phase is performed offline before the sensor nodes are deployed. In this phase, the setup server first generates two key pools: the upward key pool and the downward key pool. And the sensor nodes in the WSNs are divided with $L$ classes.

The key pool consists of $M/2$ upward key hash key chains and $M/2$ downward key hash key chains, which are divided into $L$ disjoint sub-key pools.

The upward sub-key pool: The upward sub-key pool is initiated with $M/2$ keys which are generated with a key generator. Each key of the rest classes is generated by hashing the first class key with a secure hash function. More precisely, the $j$-th class keys in the forward key pool are defined as follows:

$$U_{KL}^j = \{uk_1^j, uk_2^j, \ldots, uk_{M/2}^j\}$$

where each $uk_i^{j+1} = H(uk_i^j)$, $uk_i^0$ is the first class of the $i$-th upward hash key chain and randomly generated by a key generator.

The downward sub-key pool: The downward sub-key pool is initiated with $M/2$ keys which are generated with a key generator. Each key of the rest classes is generated by hashing the highest class key with a secure hash function. More precisely, the $j$-th class keys in the forward key pool are defined as follows:

$$D_{KL}^j = \{dk_1^j, dk_2^j, \ldots, dk_{M/2}^j\}$$

where each $dk_i^j = H(dk_i^{j+1})$ and $dk_i^{L-1}$ is the $(L-1)$-th class of the $i$-th downward hash key chain and randomly generated by a key generator.

After the two key pools have been generated, the upward and downward key rings are assigned to each sensor. To do this, the sensor nodes are divided into $L$ classes in the network and the $j$-th class sensor nodes are configured with $j$ class upward keys $uk_i^j$, and downward keys $dk_i^j$, from the upward and downward key pools. An example of key predistribution is illustrated in Figure 1.
3.2. Pairwise Key Establishment Phase

In the network bootstrap phase, each sensor node is required to broadcast its class and key ring to the neighbor nodes. Hence, each node will know which keys its neighbors have and which classes its neighbors are.

Suppose the two sensor node \( u \) and \( v \) have \( u \) \((u>0)\) upward keys and \( d \) \((d>0)\) downward keys which are in the common key chain. Assume the class two sensor node \( u \) and \( v \) are in \( l_u \) and \( l_v \) respectively, and these \( u \) upward keys are \{\( uk_{i_1}^l, uk_{i_2}^l, \ldots, uk_{i_j}^l \)\} and \( d \) downward keys are \{\( dk_{i_1}^l, dk_{i_2}^l, \ldots, dk_{i_j}^l \)\}, here \( l \) is \( l_u \) in sensor node \( u \) and \( l \) is \( l_v \) in sensor node \( v \), Without loss of generality, here we assume \( l_u \leq l_v \). Then sensor node \( u \) and \( v \) can compute the pairwise key as follows:

**Sensor node** \( u \):

\[
k_{uv} = H(uk_{i_1}^l \parallel uk_{i_2}^l \parallel \cdots \parallel uk_{i_j}^l \parallel H^{l_u-l_i}(dk_{i_1}^l) \parallel H^{l_u-l_i}(dk_{i_2}^l) \parallel \cdots H^{l_u-l_i}(dk_{i_j}^l))
\]  

**Sensor node** \( v \):

\[
k_{vu} = H((H^{l_v-l_i}(uk_{i_1}^l) \parallel H^{l_v-l_i}(uk_{i_2}^l) \parallel \cdots H^{l_v-l_i}(uk_{i_j}^l)) \parallel dk_{i_1}^l \parallel dk_{i_2}^l \parallel \cdots dk_{i_j}^l)
\]

Obviously, \( k_{uv} = k_{vu} \).

4. Performance Analysis

In this section, the proposed scheme is analyzed, including network connectivity, resilience against node capture and performance overhead.

4.1. Local Connectivity

Local connectivity is defined as the probability that two neighboring nodes can establish communication pairwise keys directly. Suppose there are two sensor node \( u \) and \( v \) in the WSNs. Let \( A(u, v) \) be the event that node \( u \) and \( v \) have at...
least one common upward key and $B(u, v)$ be at least one common downward key. From last section, we know that the local connectivity $P_{local} = P(A(u,v)) \times P(B(u,v))$.

Suppose there are $M/2$ different upward keys in each class in the upward key pool and $M/2$ different upward keys in each class in the downward key pool. And each sensor nodes store $m/2$ distinct upward keys and $m/2$ distinct downward keys. So, we have

$$P(A(u,v)) = 1 - \frac{\binom{m/2}{M/2} \binom{m/2}{M/2 - m/2}}{\binom{M/2}{m/2} \binom{M/2}{m/2}} = 1 - \frac{\binom{m/2}{M/2 - m/2}}{\binom{M/2}{m/2}}$$  \hspace{1cm} (3)$$

$$P(B(u,v)) = 1 - \frac{\binom{m/2}{M/2} \binom{m/2}{M/2 - m/2}}{\binom{M/2}{m/2} \binom{M/2}{m/2}} = 1 - \frac{\binom{m/2}{M/2 - m/2}}{\binom{M/2}{m/2}}$$  \hspace{1cm} (4)$$

Hence, the local connectivity $P_{local}$ is

$$P_{local} = P(A(u,v)) \times P(B(u,v)) = \left(1 - \frac{\binom{m/2}{M/2 - m/2}}{\binom{M/2}{m/2}}\right)^2$$  \hspace{1cm} (5)$$

**Figure 2. Network Local Connectivity $P_{local}$ for Different M and m**

Figure 2 shows the network local connectivity of the proposed scheme for different $M$. It is also noted that for larger key ring size $m$, the local connectivity also increases. From figure 1 and equation (5), we can also find the local connectivity has no relation with the class $L$. 
4.2. Resilience against Node Capture

In this section, we evaluate how the proposed scheme improves the network security in terms of resilience against node capture. We compare our scheme with some existing schemes by calculating the fraction of compromised communication among non-compromised nodes.

4.2.1. Fraction of Compromised Network Communication: Now we study the resiliency property of the proposed scheme against node compromise by calculating the fraction of communication keys in the network that are compromised due to key revealing resulted from captured nodes. Here, we calculate the probability the compromising the shared pairwise key between these two non-compromised sensor nodes when there are \( x \) sensor nodes have been captured. In the proposed scheme, the adversary can’t get a lower class key from a higher class key which is in the same upward key chain and can’t get a higher class key from lower class key which is in the same downward key chain.

Suppose \( K \) be the communication key used by two non-compromised sensor nodes. Let \( B_i \) represent the joint event that \( K \) derived from the \( i \)-th key pair \( A_i <u_{kiw}, d_{kiw}> \), \( 0<i\leq M/4 \), and \( u_{kiw} \) and \( u_{kiv} \) are both compromised. We use the notation \( K \in A_i \) to represent that “key \( K \) was derived using key pair \( A_i \)”. Let \( C_x \) be the event that \( x \) nodes has been compromised. When \( x \) nodes have been compromised, the probability of the communication key \( K \) been compromised is:

\[
P(K_{\text{compromised}} \mid C_x) = P(B_1 \cup B_2 \cup \cdots B_{M^2/4} \mid C_x)
\]

As events \( B_1, B_2, \cdots B_{M^2/4} \) are mutually exclusive and all events \( B_i \) are equally likely, we have

\[
P(K_{\text{compromised}} \mid C_x) = \sum_{i=1}^{M^2/4} P(B_i \mid C_x) = \frac{M^2}{4} P(B_i \mid C_x)
\]

Note that

\[
P(B_i \mid C_x) = \frac{P((K \in A_i) \cap (A_i \text{ is compromised} \cap C_x))}{P(C_x)}
\]

Since the event \((K \in A_i)\) is independent of the event \(C_x\) and \(A_i\) is compromised), therefore

\[
P(B_i \mid C_x) = \frac{P(K \in A_i) \cdot P((A_i \text{ is compromised} \cap C_x))}{P(C_x)} = P(K \in A_i) \cdot P(A_i \text{ is compromised} \mid C_x)
\]

As there are \( \frac{M^2}{4} \) key pairs \( A_i \), we have
\[ P(K \in A_i) = P(\text{The key } K \text{ is derived from the key pair } A_i) = \frac{1}{M^2/4} \quad (10) \]

Therefore,

\[ P(K_{\text{compromised}} \mid C_x) = \sum_{i=1}^{M^2/4} P(B_i \mid C_x) = (M^2/4)(B_i \mid C_x) \]

\[ = (M^2/4) \cdot \frac{1}{M^2/4} P(A_i \text{ is compromised } \mid C_x) = P(A_i \text{ is compromised } \mid C_x) \quad (11) \]

Now we compute \( P(A_i \text{ is compromised } \mid C_x) \). Suppose sensor node \( u \) and \( v \) are two uncompromised sensor nodes. And the level of sensor node \( u \) is \( l_u \) and the sensor node \( v \) is \( l_v \). Without loss of generality, here we assume \( l_v \leq l_u \). Assume sensor node \( c \) has been compromised, which class is \( l_c \). As there are \( L \) classes, the probability that the class of the sensor node \( c \) is less than or equal to sensor node \( v \) is \( (l_v+1)/L \), and the probability that the class of the sensor node \( c \) is greater than or equal to sensor node \( u \) is \( (L-l_u)/L \). So the probability of the any one compromised sensor node can get the upward key in any two uncompromised sensor nodes can be defined as follow

\[ P_u = \sum_{l_u=0}^{L-1} \sum_{l_v=0}^{L-1} \frac{1}{L} \sum_{l_c=0}^{L-1} \frac{l_v+1}{L} = \frac{(L+1)^2(L+2)}{12L^3} \quad (12) \]

And the probability of the any one compromised sensor node can get the downward key in any two uncompromised sensor nodes can be defined as follow

\[ P_d = \sum_{l_v=0}^{L-1} \sum_{l_u=l_v}^{L-1} \frac{1}{L} \sum_{l_c=0}^{L-1} \frac{L-l_u}{L} = \frac{(L+1)^2(L+2)}{12L^3} \quad (13) \]

So if the adversary has compromised \( x \) sensor nodes, he can get the upward keys from the \( xP_u \) compromised sensor nodes and get the downward keys from the \( xP_d \) compromised sensor nodes. Then we have

\[ P(A_i \text{ is compromised } \mid C_x) = \left( 1 - \left(1 - \frac{m/2}{M/2} \right)^{xP_u} \right) \cdot \left( 1 - \left(1 - \frac{m/2}{M/2} \right)^{xP_d} \right) \quad (14) \]

From formula (11) and (14), we have

\[ P(K_{\text{compromised}} \mid C_x) = \left( 1 - \left(1 - \frac{m/2}{M/2} \right)^{xP_u} \right) \cdot \left( 1 - \left(1 - \frac{m/2}{M/2} \right)^{xP_d} \right) \quad (15) \]
4.2.2. Comparison with Related Work: Here, we compare our scheme with EG scheme [9], and q-composite scheme (for $q=2$, 3) [10]. To access how much is the improvements gained by the proposed scheme, we compare the security performance of several schemes for different parameter situations and use the same amount of the storage per node for a fair comparison. In the comparison, we assume that each node can store 200 keys and the local connectivity $P_{local}$ is 0.34 and 0.5 respectively. Figure 4 (a)(b) clearly shows that our scheme has better security performances than that of EG schemes and $q$-composite scheme. Taken as an example, the case in which $P_{local}=0.34$, 

![Figure 3. Fraction of Compromised Communication between Noncompromised Nodes versus Number of Compromised Nodes. (a) $P_{local}=0.34$ (b) $P_{local}=0.5$](image-url)
when an adversary compromise 400 sensor nodes compromised, about be 55% of links compromised between non-compromised sensors in EG scheme will be disclosed, 79% in \( q \)-composite \( (q=2) \), 93% in \( q \)-composite \( (q=3) \) while there will only be 20% in our scheme. The reason is that in EG scheme and \( q \)-composite scheme compromised sensor nodes will disclose all the keys in them while in the our scheme only part of keys will be disclosed in the compromised sensor nodes.

![Graph of security comparison](image)

**Figure 4. Security of Comparison of EG Scheme, \( q \)-composite Scheme and our Scheme under the same Amount of Storage per Node with (a) \( P_{local}=0.34 \) (b) \( P_{local}=0.5 \)**

**4.3. Memory Usage and Communication Overhead**

*Memory usage:* Memory in sensors is often very restricted, so the key establishment protocol should use memory efficiently. According to our scheme, during the initialization
phase each sensor node needs to store \( m \) keys over finite field \( \mathbb{F}_q \). In addition, each node needs to remain the Id of the keys and the class of sensor nodes. Assume the Id of sensor nodes are chosen from a finite field \( \mathbb{F}_q \) and the class of sensor nodes are chosen from a finite field \( \mathbb{F}_q \). Thus, the overall storage overhead are \( m(\log q + \log q') + \log q' \) bits. However, in EG scheme [9] and q-composite scheme [10] the sensor nodes are need not to identity of its class, so there are more \( \mathbb{F}_q \) bits memory storage needed in the proposed scheme.

Communication Overhead: In the shared key discover phase each sensor node need to disclose of a list of \( m \) index of keys and the class of itself to its neighbor nodes. Similar to the memory overhead analysis, there are more than \( \log q' \) communication overhead needed than that of the EG scheme and q-composite scheme.

For example, if there are eight classes of sensor nodes in the system, there are will be only more 3 bits memory usage and communication overhead in the proposed scheme than in the EG scheme and composite scheme.

5. Conclusion

In this paper, a new key management scheme based on EG scheme was proposed. The proposed approach uses double one-way hash functions to generate many sub key pools. With the one-way hash function, the proposed scheme can make attackers get less key information from the compromised sensor nodes. The effectiveness of the proposed scheme has been demonstrated through analysis and comparison with other scheme.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (Grant No. 61142010).

References
