RESEARCH ON KEY PRE-DISTRIBUTION SCHEME OF WIRELESS SENSOR NETWORKS

Lingaraj.k\textsuperscript{1}, lokesh.K.S\textsuperscript{2}, Nagaveni.V.Biradar\textsuperscript{3}, Chowdari.k.k\textsuperscript{4}

\textsuperscript{1,3} Department Of CSE, R. Y. M. Engineering College, Bellary, Karnataka, India
\textsuperscript{2} Department Of ECE, R. Y. M. Engineering College, Bellary, Karnataka, India
\textsuperscript{4} Department Of CSE, Brindavan College of Engineering, Bangalore, Karnataka, India

ABSTRACT

Key management is one of the most challenging security issues in wireless sensor networks where sensor nodes are randomly deployed in a hostile territory. However, due to the resource constraints, achieving such key agreement in wireless sensor networks is nontrivial. Many key agreement schemes used in general networks, such as Diffie-Hellman and public-key-based schemes, are not suitable for wireless sensor networks. Several exiting key management schemes have been proposed in literature to establish pairwise keys for wireless sensor networks, but they either cannot offer strong resilience against node capture attacks or have overly large memory requirement to achieve high degree of connectivity. In this paper, we propose a novel pairwise key management scheme to enhancing the security. In the proposed scheme, part of keys in the key pool are computed by using hash function and the hash value are as new keys and put back into the key pool. Comparison to the existing approaches, this proposed scheme provides a stronger resilience against node capture attack.

Keywords: Wireless sensor networks, Key management, network Security.

I. INTRODUCTION

Recent advances in electronic and computer technologies have paved the way for the proliferation of wireless sensor networks (WSNs). Wireless sensor networks usually consist of a large number of tiny sensor nodes which have limited computing capability and energy resource which can be deployed anywhere and work unattended [1]. Wireless sensor networks are being deployed for wired variety of applications, including military sensing and tracking, environment monitoring, building monitoring, etc. Security becomes extremely important when sensor networks are deployed in a hostile environment. To provide security,
key management is the first things needed to. However, limited computation and energy resources of sensor nodes make it infeasible to use public key management algorithms [2].

There exist a number of key pre distribution schemes. A naïve solution is to let all the sensor nodes store a common key. This scheme does not exhibit desirable network resilience: If one node is compromised, the entire of networks will be compromised. Another solution is to let each sensor nodes carry a pairwise key with each of the rest sensor nodes. This solution can provide perfect security against node capture. However, this scheme needs much memory to store the keys when the sensor network is large. Eschenauer and Gligor proposed a probabilistic key pre-distribution scheme [3], which is referred to EG scheme in our paper. It generates a large pool of random keys and each sensor node is preconfigured a random subset of keys from a large key pool before deployment of the network. Neighboring nodes will have a certain probability to share at least one key to agree on a pairwise key for communication, two nodes find one common key within their subsets and use this key as their pairwise key. Based on the EG scheme, Chan et al. proposed q-composite scheme, which require that two sensor nodes share at least q (q>1) keys instead of just one common key to establish a secure connection [4]. The q-composite scheme increase the network resilience against node captures, but it is only advantageous when there is few captured sensor nodes. Later, Zhang et al. used Hash function to improve the capability of network to against nodes capture attack [5, 6].

Du et al. [7] proposed a pairwise key pre-distribution scheme combing the basic scheme in and Blom’s key pre-distribution mechanism together. This scheme exhibits a nice threshold property: When the number of compromised sensor nodes is less than threshold, the probability that sensor nodes other than the compromised ones are affected is close to zero. However, after more nodes compromised, almost all connections will be compromised [8,9]. Liu et al. proposed a similar pairwise key scheme based on Blundo’s polynomial-based key distribution scheme [10].

Du et al.[11] employed sensor nodes deployment in the key pre-distribution. In this scheme, multiple deployment points are identified in the sensor network and for each deployment point, a key space is pre-computed. Neighbouring deployment points have a number of keys in common. In other words, their key spaces consist of common keys. All sensors are grouped before deployment and each group corresponds to one deployment points. After deployment, sensor nodes in close neighborhood have a high probability of sharing a common key. Compare to the basic scheme [3], this scheme requires less memory to achieve an even higher connectivity because sensor nodes pick keys from a smaller subset of key pool.

The main shortcoming of the existing key pre-distribution schemes is the fraction of affected pairwise keys Increasing quickly as the number of number of compromised Nodes increases. As a result, a small number of compromised nodes may affect a large fraction of pairwise. Our scheme is based on the EG scheme. In the proposed scheme, part of keys in the key pool are computed by using hash function and the hash value are as new keys put back into the key pool. Compared with previous key pre-distribution schemes, our scheme can provide better resilience against sensor capture attack. The balance of this paper is organized as follows. Section II overviews the background of the proposed scheme in this paper. Section III describe our proposed scheme in detail. Section IV deals with the detailed performance analysis. Finally, section V offers concluding remarks.

MA system has been proven to be an efficient approach to enhance such capabilities of WSNs. Normally, the MA design in WSNs can be decomposed into four components, i.e., 1) architecture, 2) itinerary planning, 3) middleware system design and 4) agent cooperation.
Among these four components, itinerary planning determines the order of sensory data source nodes to be visited during the MA migration, which has a significant impact on the performance of the MA systems. Thus, find out an optimal itinerary for the MA to visit a number of source nodes is critical. However, finding an optimal itinerary had already been proven to be NP-hard, generally heuristic algorithms are proposed and applied to compute competitive itineraries with sub-optimal performance.

II. BACKGROUND

In this section, we overview the random key predistribution scheme proposed in [3]. This scheme consists of three phases: key pre-distribution, shared-key discovery, and path-key establishment.

1. Key Pre-distribution Phase: In this phase, a key distribution center generates a large key pool offline. Then the key distribution center selects m distinct keys from the key pool to form a key ring and loads the key ring into the memory of each sensor node.

2. Key Discovery Phase: During the key discovery phase, each sensor node broadcasts its key identifiers in clear-text or uses private share-key discovery scheme to discover the keys shared with its neighboring sensor nodes. Since all the keys are randomly selected from the same key pool, two sensor nodes may have some overlapped keys in their memories. If such a key exists, the key will be used to secure the communication link between these two sensor nodes.

3. Path Key-setup Phase: If a sensor node does not share keys with a given neighboring node, the sensor uses the key graph build during key discovery phase to find a key path to set up the pairwise key.

The size of the key pool $S$ is critical to both the connectivity and resilience of the scheme. For a given $t$, the larger the size of the key pool $S$, the lower local connectivity and the higher resilience. Local connectivity is defined as the probability that any two neighboring nodes share one key. Resilience is defined as the fraction of the secure links that are compromised after the adversaries capture a certain number of nodes.

The basic key pre-distribution scheme is the first attempt to deal with the key distribution problem in WSNs; it is more efficient than public-key based security schemes. The main problem of this scheme is it cannot provide security when the number of compromised nodes increases. Because of the low-cost hardware, sensor nodes are not tamper resistant devices. If a sensor node is captured, all its stored cryptographic information can be easily extracted by the adversary. In the basic scheme a same key may be used by different pairs of sensor nodes in a network, therefore each sensor node’s capture may be compromised the communication between non-captured sensor nodes.

III. THE PROPOSED SCHEME

Now we describe how the proposed key predistribution scheme works in detail. There are three phase in the proposed scheme: Setup Phase, Direct Key Establishment phase, and Path Key Establishment Phase. The set phase is formed to initialize the sensor nodes by distributing key information to them. After being deployed, if two sensor nodes need to establish a pairwise key, they first attempt to do so through direct key establishment. If they can successfully establishment a common key, there is no need to start path key establishment. Otherwise, these sensor nodes start path key establishment, trying to establishment a pairwise wait the help of other sensor nodes.
A. The Predistribution Phase

This phase is done offline by a Key Distributions Servers (KDS) before deploying the sensor nodes in a target field. It consists of the following steps:

Step 1: The KDS generates a very large size of key pool, which has w keys, and each key has a unique ID.

Step 2: The KDS randomly selected pw, \(0 \leq p \leq 1\), keys from the key pool. And then the KDS computes the hash value of these pw keys using a given hash function and put these derivative keys into the key pool. Suppose K is the key selected from the key pool, then \(K = H(K)\), here H is the given hash function. In this paper, we call the key generated by the KDS directly original key and the key computed by hash function derivative key. In this scheme, we suppose that the derivative K have the same key ID with the original key K.

Step 3: For each sensor node, the KDS randomly chooses t keys from the key pool S and stores them into the sensor node. Here, we need the identity of these t keys are distinct.

B. The Direct Key Establishment Phase

This phase initially takes place after the deployment of the network in the field. In this phase, if two sensor nodes want to establish a pairwise key, they need to identify a shared key. To discover whether a sensor node can establish the pairwise key directly with its neighbors, each sensor node broadcasts a list of key’s IDs and types to its neighbors. If they can use the following method to get the pairwise keys between them.

Assume the ID of the key ku in sensor node u is the same as that of the key kv in sensor node v. If the key ku and kv are both original keys or derivative keys, they can use the ku or kv as their communication key. If the key ku is a original key and the key kv is a derivative key, the sensor node u and v can use the ku=H(kv) as their communication key, and vice versa.

C. The Path Key Establishment Phase

If direct key establishment fails, two sensor nodes will have to start to establishment a pairwise key with the help of other sensor nodes. To establish a pairwise key with sensor node j, a sensor node i needs to find a path between them such that any two adjacent sensor nodes in the path can establish a pairwise key directly. Then either sensor node i or j initiates a request to establish a pairwise key with the other sensor node through the intermediate nodes along the path.

IV. PERFORMANCE ANALYSIS AND COMPARISON

To evaluate the security property and networks performance of the proposed scheme, we compare the proposed scheme with the exiting key predistribution schemin this section. We present our analytical result on the following two metrics: local connectivity and resilienc against node capture.

A. Local Connectivity

Local connectivity is an important metric to evaluate key predistribution scheme. To achieve desired glob connectivity, the local connectivity must be higher than certain threshold value called the required local connectivity.

Now we calculate the local connectivity \(P_L\). From the last part, we know that if the sensor nodes have a common key ID, they can establish a pairwise key directly. So
PL=1- Pr(two nodes do not share any common key identity), we have

\[
P_L = 1 - \frac{\binom{w}{m} \binom{w-m}{m}}{\binom{w}{m} \binom{w}{w}}
\]

(1)

Here, \( w \) is the size of the key pool \( P \). Fig.1 show the probability to establish direct key given different number of the keys preload in each sensor.

**B. Resistance against Node Capture**

Nodes deployed in hostile environments are prone to capture. Given some nodes captured, adversaries can compromise pairwise key derived from polynomial stored in these nodes. If an adversary compromise more than \( t \) sensor nodes which use a common polynomial to generate pairwise key, he will compromise this polynomial. With the compromised polynomial, the adversary can eavesdrop not only the connections linked with the compromised sensor nodes, but also other additional ones secured by those keys which generated by the compromised bivariate polynomial. We measure the resilience against node capture by the fraction of all connections that adversaries can compromise over the total number of connections of the network given some nodes captured.

Suppose the sensor node \( u \) and node \( v \) are two non-compromised sensor nodes. Now we study the probability the communication pairwise key between these two non-compromised sensor nodes when there are \( x \) sensor nodes have been captured. As the \( t \) keys are randomly from the key pools, so the probability of key being original key \( w/w+p \) is of being derivative key is \( pw/w+pw \). From the mechanism of setup phase, we know that the original key in the compromised sensor nodes will disclose the original and derivative keys with the same identity and the derivate key can only disclose the derivative keys with the same identity.
Let the number of the total captured sensor nodes be $x$. Hence, we have the probability $P$, that any secure link between two uncompromised sensor nodes is compromised when $x$ sensor nodes have been captured is

$$P_b = 1 - \left(1 - \left(\frac{1}{1 + p} \cdot \frac{m}{w} + \frac{p}{1 + p} \cdot \frac{p_m}{w}\right)^x\right)$$  \hspace{1cm} (2)$$

Figure 2 includes that the relationship between the fractions of compromised links for non-compromised nodes and the number of compromised nodes. We can see that the more number of sensor nodes compromised, the higher the fraction of compromised links.

C. Comparison with Previous Schemes

Now let us compare our scheme with the basic probabilistic [3], the q-composite [4]. Figure 3 shows the security performance of our scheme, the basic probabilistic scheme [3], the q-composite scheme, and the scheme [4]. There figures clearly show that before the number of the number of compromised sensor nodes reaches threshold, our scheme performs much better than the basic scheme and q-composite scheme. For example, under the local connectivity is 0.33, in our proposed scheme when there 400 compromised sensor nodes, the fraction of the compromised link is 45%, the fraction of the compromised in EG scheme is 55%, that of in q-composite (q=2) is 79%, that of in q-composite (q=3) is 94%.
V. CONCLUSION

In this paper, we addressed the problem of itinerary planning for multi-agent based data dissemination, facilitating concurrent sensory data collection to reduce task duration extensively. The proposed multi-agent route scheduling algorithm has the similar complexity with most of single agent based route scheduling algorithm, and can be flexibly adaptive to network dynamics in various network scales. We will propose more efficient source-grouping algorithm in our future work.

In this paper, we proposed a robust key pre-distribution scheme, which is based on the EG scheme. In the proposed scheme, part of keys in the key pool are computed by using hash function and the hash value are as new keys put back into the key pool. Compared to exiting key predistribution schemes, our schemes is substantially more resiliency against sensor nodes capture.

REFERENCES


