무선센서네트워크에서의 강화된 키 관리 프로토콜

(An Enhanced Key Management Protocol For Wireless Sensor Networks)

Binh Thanh Dang (Sejong Univ, Master student)
Seok Lee (KIST, Senior Researcher)
Hyung Seok Kim (Sejong Univ, Professor)

Key Words : Key Management, Partial Key, Group Key, Wireless Sensor Network

목 차

I Introduction

Wireless sensor networks (WSNs) has become a widely researched subject in the industry. A WSN is a low-cost network including thousands of nodes detecting physical phenomena. WSNs have a wide range of applications, including military, environment, health, home automation, manufacturing, etc. Sensor nodes are battery powered, limited in memory size and computational power. These limitations lead to many problems, including security implementation.

Security is one of the most important aspects in researching WSNs. When we look at the security aspect of a WSN, many well-known approaches become unsuitable for they were designed without security considerations. However, many applications need the sensed data to be exchanged securely. There are several works on WSNs security have been published.

Using keys to secure exchanged data in WSNs has been studied in some years. Before the WSNs can securely transmit the data, encryption keys must be established among sensor nodes.

Several approaches for key management have been discussed. Panja et al [1] proposed a dynamic key management scheme for tree-based hierarchical sensor networks. This protocol has many advantages in comparison with others, especially in large WSNs. However, their approach is not perfect. Keys need to be managed in an efficient way since keys are generated, distributed and in cases when a new node joins the network or a key needs to be deleted (for example, when a compromised node is detected and its key needs to be removed). Panja's scheme uses a dynamic approach, where keys can be granted dynamically. While Panja focused on updating group keys, the cases of key revocation and node addition still aren't addressed explicitly. Although these problems can be solved by re-computing group keys and partial keys, this action still requires a lot of computational power.

Another problem with Panja's scheme is partial keys on leaf nodes seem to be simple. This could lead the network to be vulnerable when it has to cope with attacks.

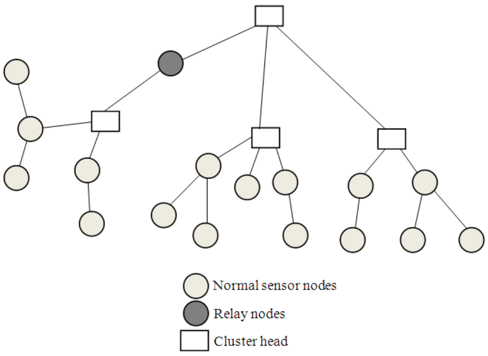
In this paper, we propose an enhanced version on Panja's scheme in order to solve those problems. The following parts of the paper are organized as below. The related key management protocol will be discussed in Section 2. Section 3 will introduce our proposed approach. Conclusions will be presented in Section 4.

II Related Work

Panja et al [1] introduced a solution for key management in WSNs. Instead of using a key pre-distribution scheme, they proposed a dynamic approach, where keys can be calculated dynamically. The network is organized in a hierarchical infrastructure, as described in Fig.1. Nodes are grouped into clusters. Each cluster has its own cluster head. There will be nodes used as Head of Cluster Heads.

There are two main types of keys: partial keys and group keys. With leaf nodes, partial keys are chosen as a random number. Partial keys at other levels in the tree are determined by using a function embedded in the sensor nodes. Each node uses partial keys of their children as inputs to the function $\alpha^{k_1 \oplus k_2} \bmod p$ where p is a prime number, k_1 and k_2 are two children's partial keys and a is a primitive root of p.

Group keys are calculated by using a bottom-up approach. Partial keys from lower levels are collected up to the root node to form the group key. There are two different types of group keys: group key for intra-cluster communication and group key for inter-cluster communication.



<Fig 1> Network architecture

By using this approach, number of keys needed to be stored in each sensor nodes is smaller than in other ones. Also, the fact that keys can be changed frequently and dynamically helps reducing the probability that nodes in the network may be captured.

However, the low complexity of the leaf nodes' partial keys may be this protocol's Achilles' heel. Though it's dynamic approach can reduce the probability when the nodes are captured, the weak partial keys may be easily compromised by an adversary.

III Our Approach

We made some modifications to Panja's scheme to increase its security, as described below.

1. Partial keys computation

Instead of using random numbers for leaf nodes' partial keys, we use a pre-distributed key pool. This solution is taken from Eschenauer et al's solution [2]. First of all, a large pool of keys is generated. Afterwards, k keys is randomly taken out of this pool, $k<<N$, where N is the number of nodes in the network. Each node will receive its own key ring.

As in Panja's approach, we assume that after network organization, the cluster head knows its members' position *Pos(l,v)*, where l is the node's level from the cluster head and $v$is the position of the node from the left. Firstly, the cluster head sends a message to its leaf nodes to ask them to create their own partial keys. Each leaf node will choose its own key from its key ring. Then, it send its partial key to the parent node. The parent node computes its partial key by using Panja's partial key computational algorithm. A pre-deployed symmetric key will be used to encrypt /decrypt the partial keys here.

2. Group keys computation

Firstly, a pre-deployed symmetric key will be used to encrypt /decrypt the partial keys and the group key. After the group key is computed, it will be used for these purpose. The pre-deployed symmetric key is discarded then. Panja's strategies will be used to compute group keys. Partial keys of nodes under the cluster head will be sent to the cluster head. The cluster head will compute the group key. Then, the group key will be broadcasted in the cluster.

3. Node addition

When a new node is accepted to join a cluster, it will be a leaf node. The cluster head sends a message to this node to require it to calculate its partial keys. The group key will be update after this process.

4. Key revocation

Key revocation may occur in key management process, especially when a node is found compromised. The cluster head of the cluster where the compromised node is in sends a message to all its members (except for the node needed to be revoked) to update the group key. When a new group key is created, the compromised node will not be able to communicate with others. If this node is recovered and want to join the cluster again, it will be treated as a new node.

IV Conclusion

The tree-based hierarchical structure of the network in Panja's scheme is very scalable. In this paper, we proposed an enhanced version of his protocol for better robustness.

References

[1] B. Panja, S. K. Madria, and B. Bhargava, "Energy and Communication Efficient Group Key Management Protocol for Hierarchical Sensor Networks," SUTC '06: Proc. IEEE Int'l. Conf. Sensor Networks, Ubiquitous, and Trustworthy Comp., 2006, pp. 384-393.

[2] L. Eschenauer and V. D. Gligor, "A Key-Management Scheme for Distributed Sensor Networks," Proc. 9th ACM Conf. Comp. and Commun. Sec., 2002, pp. 41-47.