Cluster Head Monitoring Nodes Secret Selection Protocol in Sensor Networks

Jinhui Yuan1,*, Hongwei Zhou2, *
1Zhongyuan University of Technology, Zhengzhou, China
2Information Engineering University, Zhengzhou, China

*Corresponding author e-mail: hong_wei_zhou@126.com, jcyjh@126.com

Abstract. In sensor networks, as the data collection point of nodes in the cluster, cluster head is easy to be the key target of attack. Once the cluster head node is controlled by the attacker, it will have a fatal impact on the security of the network. In order to solve the above problem, this paper proposes a cluster head monitoring nodes secret selection protocol. It uses secret selection algorithm to select the cluster head monitoring node from the neighbor nodes of the cluster head node. Thus, the attacker can not confirm the monitoring node, it results that the cluster head node can not be attacked. The protocol uses the spatio-temporal correlation between cluster head and cluster head monitoring nodes. Once it finds that cluster head node is suspected anomaly, it reports immediately to sink node. The analysis of protocol security shows that the protocol can meet the energy efficiency of sensor network and ensure the security of the system.

1. Introduction

Sensor networks have been widely used in all aspects of society. Due to the manufacturing cost, it is impossible to deploy complex security protection mechanism in the node itself. This makes sensor nodes unable to defend against security attacks independently and easy to be controlled by attackers. Secondly, sensor nodes are often deployed in environments that are difficult for managers to fully control, such as battlefield frontiers, volcanoes, forests etc., which makes sensor nodes vulnerable to attackers' access and control. At present, the security of sensor networks has caused great concern.

In sensor networks, cluster-head nodes often become the primary target of attackers. In order to reduce the energy consumption of sensor nodes, people often divide sensor networks into several clusters. Each cluster node submits perceived data to the cluster head node. The cluster head node uses some algorithm to process the data and then submits it to the upper node. In some schemes, when sensor node believes that there is no significantly change in the sensing data, it is no longer submit data to cluster head node. And the cluster head node reports its data according to the default rule. Due to the importance of cluster head node in the routing, the attacker naturally attacks the cluster head node first, and obtain the data submitted by the entire cluster by controlling the cluster head nodes. This will lead to more threats to the security of sensor networks.

In order to discover this behavior, the neighbor nodes of the sensor node can observe whether the data sent by the node is normal. Because the target node and the detection node may have some data connections such as temporal and spatial correlation, the detection node can detect the tampering
behavior of the attacker according to the change of the data, so that the administrator of the sensor network can be notified in time. For example, an attacker controls a node in the network and sends fake data. In order to discover this behavior, the neighbor nodes of the sensor node can observe whether the data sent by the node is normal. For there are spatial and temporal correlation among the target node and the detecting node, the detecting node can detect the tampering behavior of the attacker according to the change of the data, thereby timely notifying the administrator of the sensor network.

There are generally two architectures to implement anomaly event detection. One is done by the node itself. According to the inconsistency between the collected data and historical data, the detection mechanism detect anomaly event to inform the superior node in time. However, since the attacker may directly attack the cluster head node, the anomaly detection mechanism based on itself has obvious defects: when the controller controls the cluster head node, the detection mechanism completed by the cluster head node also fails. To this end, people use the method of neighbor node cooperation to complete anomaly event detection. The general process is as follows. All the neighbors of cluster head nodes detect the possible anomalies of cluster head node according to the principles of temporal and spatial correlations. By collecting the opinions of all neighbors, it can be found whether the cluster head is anomaly. This scheme can effectively avoid the attacker's influence on the detection mechanism itself. However, for a large number of nodes participate in the detection, it will affect the energy overhead of nodes. It may lead to excessive energy consumption of sensor nodes, thus resulting in a decline in the life cycle of sensor network nodes.

To this end, we propose a secret selection protocol for cluster head monitoring nodes. Our motivation is to secretly select some nodes as cluster head monitoring nodes from neighbor nodes of cluster head nodes, and to monitor cluster head nodes under the premise that the attacker is unknown. If all neighbor nodes of the cluster head participate in anomaly detection, it will consume too much energy. In order to avoid this, we want to extract some nodes to perform anomaly detection according to certain principles, while other neighbor nodes are latent. The key of cluster head node secret selection protocol is that, unless the cluster head monitoring node is selected, neither cluster head node nor neighbor node knows which nodes will act as the monitoring node in the next round of monitoring cycle. To the best of our knowledge, we are the first selection protocol for cluster head monitoring nodes.

We use the following methods to realize the secret selection of cluster head monitoring nodes. Each sensor node randomly apply for several monitoring tokens at the beginning of each round of monitoring period; when the cluster head node is officially selected, each candidate node participating in the monitoring randomly broadcasts a small piece of text; each node collects these text fragments, and the serial number will be assembled into a complete text; each node computes the hash values of these words and performs modular operations to obtain the on-duty token for the next round of monitoring period; when the on-duty token matches the token held by the node, it means that the current node is selected as the cluster head monitoring node for the next monitoring cycle; each monitoring cycle, the cluster head monitoring node is re-selected. Focusing on the goal of cluster head node secret selection, we propose a cluster head node secret selection protocol.

We analyze and discuss the Cluster Head Monitoring Nodes Secret Selection Protocol. Firstly, on the secrecy of cluster head nodes, we consider that even if the attacker controls the cluster head nodes and a small number of neighbor nodes, it is difficult to fully know the next round of monitoring node information, unless the attacker controls more neighbor nodes. This undoubtedly increases the difficulty of attack. It should be noted that we are not trying to completely eliminate attacks. We just want to use this protocol to improve the difficulty of attack. Secondly, we analyze the cluster head surveillance service. Although there may be a risk of monitoring holes, this risk can be effectively reduced by increasing the number of tokens. We also illustrate the difficulties and problems that may exist in the implementation of Cluster Head Secret Selection Protocol with our application cases.
The rest of the paper is organized as follows. Section 2 presents our motivation. Section 3 proposes attacker model. Section 4 presents Cluster Head Monitoring Node Secret Selection Protocol. Section 5 is our discussion. Related work is in section 6. At last, we conclude this paper.

2. Motivation

In sensor networks, cluster-head nodes are the hotspot of attack. In order to reduce the energy consumption of sensor nodes, sensor networks are divided into several clusters. The cluster head nodes collect the data perceived by the nodes in the cluster. In order to discover attacks on cluster-head nodes, people often use neighbor nodes of cluster-head nodes to participate in anomaly detection. In this way, in order to avoid the discovery of attacks, attackers need to control cluster head nodes and all their neighbors, which increases the difficulty of attacks.

However, some scenarios are not suitable for multi-node cooperative detection mode. In our application case, the sensor nodes are densely deployed. For a single cluster head node, there may be dozens or even hundreds of neighbor nodes. If a large number of neighbor nodes are involved in anomaly detection, the energy of these neighbor nodes will be wasted. Therefore, we consider selecting a part of nodes from a large number of neighbor nodes to participate in anomaly detection, while other neighbor nodes are latent, thereby saving energy costs.

In order to achieve the goal, an important prerequisite is that the attacker cannot know which nodes are currently monitoring. If the attacker can know the nodes involved in anomaly event detection, then the attacker can control these nodes purposefully, thus reducing the difficulty of the attacker's attack. To avoid this, it is necessary to ensure that the attacker is unaware of the information. At the same time, the node participating in anomaly event detection can not obtain this information, because once the neighbor node acquires this information, the attacker can also obtain the node information participating in anomaly event detection by controlling the neighbor node.

3. Attacker Model

Because of the limitations of sensor nodes, it is not difficult for attackers to control sensor nodes. In this paper, we make the following assumptions.

1. The attacker can successfully control a small number of nodes in the sensor network.
2. The attacker can control any node including cluster head node.
3. The attacker can successfully control cluster head nodes without being detected by security mechanism only if he controls all neighbor nodes involved in anomaly detection.
4. The total number of nodes that an attacker can control is not too large, which affects the overall security of sensor networks.

In our discussion, we analyze the number of attackers controlling nodes and their attack success rate.

4. Cluster Head Monitoring Node Secret Selection Protocol

4.1. Overview

Firstly, we discuss the scenarios adapted to cluster head monitoring node secret selection protocol. There are two obvious features in the application background of cluster head surveillance node secret selection protocol as follows.

1. Sensor nodes are dense deployment in the networks.
2. It is necessary to deploy anomaly detection system for cluster head nodes.

For example, for sensor networks used in military surveillance or precious goods surveillance scenarios, people are willing to deploy dense sensor nodes. And when sensor networks select cluster-head nodes, they hope that the neighbor nodes of cluster-head nodes will monitor the trustworthiness of cluster-head nodes' behavior. When there are more neighbor nodes of cluster head nodes, in order to reduce the energy cost of neighbor nodes, some neighbor nodes can be selected secretly to be responsible for monitoring.
Then, the design goal of cluster head node secret selection protocol is presented. Cluster head node secret selection protocol first ensures that the selected node is not known by the cluster head node or other neighbor nodes which may be controlled by the attacker. This is the only way to ensure that an attacker cannot easily bypass the surveillance. Secondly, the cluster head node secret selection protocol must adapt to the dynamic joining and exiting of neighbor nodes. In most node deployment scenarios, it is not possible to ensure that nodes are immovable or unchangeable. Thirdly, even if the attacker controls the cluster head nodes and a small number of neighbor nodes, it can not affect the implementation of cluster head monitoring behavior. Finally, the cluster head node secret selection protocol can not require sensor nodes to perform frequent computation or communication, which will increase the energy overhead of sensor nodes and reduce the whole network life cycle.

We summarize the design objectives of cluster head node secret selection protocol as follows.

1. In addition to the selected nodes, the cluster head node and the candidate node can not know the nodes that monitor the cluster head node.
2. Adapt to the situation where the candidate node dynamically increases or decreases.
3. If cluster head nodes or a small number of candidate node are attacked, it does not affect the implementation of cluster head monitoring behavior.
4. Complex computation and frequent communication are not required.

Guided by the above design objectives, we propose the cluster head monitoring node secret selection protocol. After each monitoring period is completed, the protocol secretly selects some neighbor nodes to perform the next round of monitoring on the cluster head node.

4.2. Cluster Head Monitoring Node Secret Selection Protocol
We describe the operation sequence of cluster head monitoring node secret selection protocol as follows.

1. After the node is selected as the cluster head node, it determines the node set that participates in the cluster head monitoring node selection, and the nodes in the set are candidate nodes.
2. Each node establishes its own token set after identifying the identity of the candidate node.
3. The node with the minimum sequence number in the candidate node set, determines the node responsible for the cluster head node monitoring task in the next round of monitoring period. During the selection process, even if the attacker controls the cluster head node or a small number of candidate nodes, the cluster head monitoring nodes cannot be obtained in the next round.
4. During the monitoring period, if the new sensor node is deployed near the cluster head node, it becomes the neighbor node of the cluster head node. Then it is added to the candidate node set.
5. During the monitoring period, if a node is unable to perform the monitoring task due to failure or other reasons, the node is removed from the candidate node group.
6. At the end of the monitoring period, the node with the minimum serial number in the node group checks whether it has reached the group maintenance period. If it is, it determines the status of the candidate node group and maintain it.
7. At the end of the monitoring period, a new token is generated for each candidate node, and its token set is updated.
8. At the end of the monitoring period, the node with the minimum sequence number in the node group determines the node responsible for the cluster head node monitoring task in the next monitoring period.
9. During the monitoring period, the protocol repeats (4) - (8) steps. If the cluster head node no longer acts as the cluster head, the cluster head node revokes the node group and the protocol terminates.

5. Security analysis
The fundamental feature of this protocol is to ensure the concealment of cluster head monitoring results. The attacker cannot know the result of the selection even if it controls the cluster head node or a small number of candidate nodes. We analyze three scenarios respectively, in which the attacker can
not get the selection result of cluster head monitoring node. In the first attack scenario, the attacker does not control any nodes and can only listen to data transmitted over the wireless network. In the second attack scenario, the attacker has already controlled the cluster head node. In the third attack scenario, the attacker not only controls the cluster head node, but also controls a small number of candidate nodes.

We first analyze the first attack scenario. In this case, the attacker can collect the secret selection messages sent by all nodes and get the randomly generated strings. This means that the attacker can calculate the work token for the next monitoring period. However, since the attacker does not control any nodes, he will not be able to know the set of tokens held by each node. The node that ultimately undertakes the monitoring task must be the node that holds the hit working token. When the attacker cannot know the secret token held by all the nodes, he cannot know the next round of cluster head monitoring nodes. An attacker who simply relies on sniffing cannot break this agreement.

Then, we analyze the second attack scenario. In this case, the attacker can send a fake node secret selection message through the controlled cluster head node. However, although the controller can control the character fragments generated by the cluster head node itself, he still cannot control the final working token. For in the process of computing the working token, we use the HASH algorithm. Even if the attacker can arbitrarily tamper with some part of the algorithm input, it still cannot control the output of the algorithm. In addition, the cluster head monitoring node monitors the cluster head node, so the cluster head node can forge messages, but it must send all kinds of messages according to the protocol. For example, in the node addition subprotocol, cluster head node is the key node and needs to send relevant messages. Whether the message is timely and compliant, the cluster head monitoring node will check it. It is impossible for an attacker to control only the cluster head node to destroy this protocol.

Finally, we analyze the third attack scenario. In this case, the attacker has already controlled the cluster head node and a small number of candidate nodes. However, the attacker still cannot get the result of the working token. As long as there is a legitimate candidate node, the attacker cannot control the input of the HASH algorithm completely. According to the characteristics of the HASH algorithm, the attacker cannot control the output of the HASH algorithm. Since the token generation is performed by each node alone, even if the attacker controls a small number of candidate nodes, the token generated by the other nodes cannot be known. Therefore, an attacker who only controls a small number of nodes cannot destroy this protocol.

In summary, the design goals of this agreement are achieved. The attacker cannot know the monitoring node of the next monitoring period. However, this agreement cannot completely defend against denial of service attacks. For example, after the attacker controls the cluster head node, the cluster head node is required to stop sending related messages, which will make the protocol unable to add new nodes. However, this protocol only requires the detection of attacks, not a complete defense against the attack. We believe that it is beyond the scope of this paper, we will continue to improve it in the future work.

6. Related Works

In sensor networks, clustering algorithm [1] is an importance energy-efficient data gathering technique. According to readings of sensor nodes, [2] divides them into several clusters. In order to transmit data to sink node, it constructs routing trees. [3] proposes LEACH(Low Energy Adaptive Clustering Hierarchy), which is a routing protocol for wireless sensor networks. In order to reduce the energy consumption and improve the overall lifetime of the network, cluster head nodes are selected randomly in a circular way, and the energy load of the whole network is evenly distributed to each sensor node. [4] proposes an optimized clustering technique based on spatial-correlation in sensor networks. Through utilizing similarity degree to construct clusters, it can get rid of unexpected data and get protected approximate results.

Meanwhile, it is energy-efficient for only cluster head transmitting data to sink in the network. [5] considers that existing clustering algorithms build the routing structures according to the sequence of
events happening. As a result, routing trees of the network is low quality, and the network load is unbalanced. They propose YEAST (dYnamic and scalablE tree Aware of Spatial correlaTion) to solve the above problems. [6] proposes DDCD(Data Density Correlation Degree) clustering method based on the data density correlation degree, which is a spatial correlation measurement between sensor node and its neighbor nodes. As a result, the distortion of representative data to its correlated data is very low in sensor networks. And, the clusters shape is adapted to the real environment. [7] proposes a characteristic model to derive statistically whether the behavior of sensors is normal. It can detect the inconsistencies and redundancies between nodes, and also greatly reduce the total amount of sensing data. [8] considers that sink node usually receives outdated data while results to high delays in current solutions. They propose EAST (Efficient data collection Aware of Spatio-Temporal correlation) algorithm, which sends the gathered data to sink node utilizing the shortest path. And then, it reaches near real-time data collection making full use of the spatio-temporal correlation in sensor networks.

However, none of the above algorithms has the ability to select cluster head monitoring nodes, and the selection results are public to both users and attackers. The existing algorithm is convenient for users to choose, but it is also more convenient for attackers to determine the target.

7. Conclusion

In this paper, we propose the cluster head monitoring nodes secret selection protocol. Cluster head monitoring nodes and cluster head node are neighbor nodes, which can receive some or all of the data that cluster head node receive. The cluster head monitoring nodes use secret selection algorithm, which makes it impossible for the attacker to confirm the monitoring nodes and then attack the cluster head node (of course, the attacker can attack all the neighbor nodes of the cluster head node, but this will undoubtedly increase its attack difficulty). The cluster head monitoring nodes use the temporal and spatial correlation between the cluster head node and them, which find the anomaly of suspected cluster head node and then report it immediately. The sink node collects the exception report, comprehensively evaluates whether the target cluster head node is abnormal, and determines the subsequent actions. The security analysis of protocol shows that the protocol can meet the energy efficiency of sensor network and ensure the security of the system.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments that helped improve the presentation of this paper. The work is supported in part by the National Natural Science Foundation of China (61303074), Henan Province Science and Technology Research Project (182102210589, 192102210274), Information Security Technology Key Laboratory Open Fund (KJ-15-105).

References

[1] J. Al-Karaki, and A. Kamal. “Routing techniques in wireless sensor networks: A survey,” IEEE Wireless Communications, vol. 11, no. 6, pp. 6-28, 2004.
[2] S.Yoon, and C.Shahabi, “The Clustered Aggregation (CAG) technique leveraging spatial and temporal correlations in wireless sensor networks,” ACM Transactions on Sensor Networks, vol.3, no.1, pp: 3:1-39, 2007.
[3] Abbas Nayebi, Hamid Sarbazi-Azad. Performance modeling of the LEACH protocol for mobile wireless sensor networks [J]. Journal of Parallel and Distributed Computing, 2011, 71 (6).
[4] Yuan J, Chen H . THE OPTIMIZED CLUSTERING TECHNIQUE BASED ON SPATIAL-CORRELATION IN WIRELESS SENSOR NETWORKS [C]// IEEE Youth Conference on Information, Computing & Telecommunication (YC-ICT2009)
[5] Leandro A. Villas, Azzedine Boukerche,Horacio A.B.F. de Oliveira, Regina B. de Araujo,Antonio A.F. Loureiro. A spatial correlation aware algorithm to perform efficient data collection in wireless sensor networks [J]. Ad Hoc Networks, 2014, 12.
[6] Yuan F, Zhan Y , Wang Y. Data Density Correlation Degree Clustering Method for Data
Aggregation in WSN [J]. IEEE Sensors Journal, 2014, 14 (4): 1089-1098.

[7] Kulau U, Breuer T, Wolf L. A Clustering-Based Characteristic Model for Unreliable Sensor Network Data [C]// 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT). IEEE, 2015.

[8] An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks [J]. Computer Communications, 2013, 36 (9): 1054-1066.