Applied Mathematics & Information Sciences 5 (2) (2011), 97S-103S
– An International Journal

A Secure Mechanism for Data Collection in Wireless Sensor Networks

Yuxin Mao
School of Computer and Information Engineering, Zhejiang Gongshang University,
Hangzhou 310018, P.R.China
Email Address: maoyuxin@zjgsu.edu.cn

Received October 02, 2010; Revised November 13, 2010

Due to some intrinsic features of wireless sensor networks it is difficult to perform efficient intrusion detection against malicious nodes in such a resource-restricted environment. We propose a novel secure mechanism for wireless sensor networks. The mechanism consists of a key-based secure routing algorithm and a counter-based intrusion detection algorithm. The approach is able to protect the data collection in a wireless sensor network even if there are malicious nodes in the network. By comparison with existing research efforts the proposed approach is easy to be implemented and performed in resource-constrained wireless sensor networks.

Keywords: Data Collection; Encryption; Intrusion Detection; Wireless Sensor Network

---

1 Introduction

An Intrusion Detection System (IDS) monitors a host or network for suspicious activity patterns outside normal and expected behavior [1][2]. Currently there are a number of research initiatives on intrusion detection for wireless sensor networks (WSNs). Although intrusion detection is an important issue to WSN, the research on intrusion detection in WSNs is still preliminary [3]. Due to the unattended nature of sensor nodes in WSN it is possible for an adversary to compromise some nodes and crack the detail of the node including the embedded secure mechanism. Therefore no matter how complete the secure mechanism is, as long as it is revealed to the malicious nodes, the adversary can take action to destroy the network against the secure mechanism.

In this paper we present a novel secure mechanism for WSNs, which is composed of a secure routing algorithm and an intrusion detection algorithm. The major contribution of this paper is that we propose an original secure mechanism to defend WSNs against both tampering and packet-dropping attacks, for the first time. The approach is able to protect the data collection in a WSN even if some sensor nodes are compromised by an adversary. We provide a relatively simple but reliable approach to support secure data collection in WSN. The routing and network layer of WSNs is threatened by various attacks. In this study we mainly focus on the attacks with tampering and packet dropping (e.g. selective forwarding) in the network and routing layer of WSN.

2 Key-Based Secure Routing for WSN

In this paper we consider a relatively simple WSN. Each sensor node in the WSN is battery-powered and has limited sensing, computation and wireless communication capabilities. Each node has a unique identity in the network. The sink is a data collection center equipped with sufficient computation and storage capabilities. Sensor nodes generate sensor information and aggregate data packets. The sink collects data from sensor nodes periodically. The sink is regarded as trustworthy. We only focus on secure routing between sensor nodes and the sink. An adversary is able to compromise a node or even physically capture a node. However, the sink is not going to be compromised and we refer to a sensor node not being compromised when we talk about the source node of routing. We assume that malicious nodes, in order to allay suspicions, selectively drop only a small proportion of all packets passing by rather than every packet.

We assume that each node is preconfigured with a unique and symmetric key K that it shares only with the sink before deployment. This key K is used to encrypt sensor data and generate MACs (Message Authentication Codes) for the data. In order to achieve extra security, instead of using the key K directly, each sensor node can derive a separate encryption key K_E and a MAC key K_M from the shared key K [4]. Two communicating nodes A and B share a unique encryption key $K_{A,B}$ and a MAC key $K'_{A,B}$.

On the basis of the network model, we can perform secure routing from a source node to the sink. The security mechanism of routing is mainly based on the SPINS protocol. We illustrated the key-based secure routing algorithm as follows:

Algorithm 1 Key-based secure routing algorithm for WSN

Input: A WSN with a collection of sensor nodes $\overline{S} = \{S_0, S_1, \ldots, S_n\}$, a source node S_0, and a sink node S_k, where $S_0, S_k \in \overline{S}$.

if S_0 wants to send a data packet D to S_k

  S_0 **selects** its next hop S_t ($S_t \in \overline{S}$) from its neighbors

  S_0 **sends** the encrypted packet E to S_t

  $E_0 = [\{D \| CNT\}_{<K_0,C>}, MAC(K'_{0,t}, C \| \{D \| CNT\}_{<K_0,C>})]$

  CNT++

  for each intermediate node S_t

    S_t **receives** the data packet E_{t-1} derived from S_{t-1}

S_t **extracts** and **records** CNT from E_{t-1}

S_t **forwards** E_t to S_{t+1}

$$E_t = [\{D \| CNT\}_{<K_0, C>}, MAC(K'_{t,t+1}, C \| \{D \| CNT\}_{<K_0, C>})]$$

end loop

end if

if S_k **receives** the data packet E_m derived from S_0

S_k **extracts** and **records** CNT from E_m

S_k **decrypts** E_m into D

end if

Here K_i is the encryption key for S_i and the sink S_k, $K'_{i,j}$ is the MAC key for nodes S_i and S_j. In this algorithm the source node attaches each data packet with a sequential number that is continuous for every data packet sent out. We use a counter CNT to represent the sequential number for data packet. Other than encrypt the content of the data packet, we also encrypt the value of CNT to prevent it from being destroyed via routing. As the data packet from a source node to the sink is encrypted by using K_E and verified by MAC, it is difficult for a malicious node to tamper the data packets as they pass by the node. A tampering behavior is detected by the sink immediately. If the malicious node decides to drop the data packet instead of tampering with it, the sink with an embedded intrusion detection mechanism is able to detect its behavior immediately. The details about the intrusion detection algorithm are illustrated in the following sections.

3 Intrusion Detection Algorithm

In this section we present a counter-based intrusion detection algorithm for WSN with malicious nodes. The algorithm is an online detection algorithm. It is able to detect malicious nodes after some malicious attacks occur during the process of data collection. The intrusion detection algorithm is illustrated as follows:

Algorithm 2 Counter-based intrusion detection algorithm for WSN

Input: A WSN with a collection of sensor nodes $\overline{S} = \{S_0, S_1, \ldots, S_n\}$, a source node S_0, a sink node S_k and a collection of malicious nodes $\overline{S}_m = \{S_i, S_{i+1}, \ldots, S_j\}$, where $S_0, S_k \in \overline{S}$, $\overline{S}_m \subset \overline{S}$.

Output:

S_0 **sends** a series of data packets $\overline{D} = \{D_1, D_2, \ldots, D_m\}$ to S_n with a time interval of Δt

for each intermediate node S_{mi} on a routing path from S_0 to S_k

```
  S_mi caches the latest three packets passing by
end loop
for each pair of packets (D_i, D_i+1) S_k receives
  S_k decrypts their contents by key
  if S_k detects a tampered packet
     S_k broadcasts an alert packet
  end if
  S_k verifies their sequential numbers
  if S_k detects a discontinuous sequential number
     S_k broadcasts an alert packet
  end if
  for each intermediate node S_mi receiving the alert
     S_mi verifies the packets within its cache
     if S_mi detects a missing packet
       S_mi sends back an alert to S_k
     else if S_mi detects a tampered packet
       S_mi sends back an alert to S_k
     else
       S_mi sends back a normal response packet
     end if
  end loop
  if S_k receives a collection of response packets
    if an intermediate node S_mi does not send back a response
      S_k records the identity of S_mi
    end if
    S_k analyzes the status information of the nodes on the routing path
    S_k finds out the malicious nodes
    S_k broadcasts the identity of malicious nodes
  end if
end loop
```

As we have mentioned above, there are two methods a malicious node can attack the WSN, tampering with the data packet or dropping it. We can show that the proposed algorithm is able to deal with both of them.

When the sink receives a collection of response packets from the nodes on the routing path, it tries to analyze these packets to detect malicious nodes. We can denote the status of a node in the response stage by *status bit*. We denote the status that a node replies a negative packet by 1, the status that a node replies a positive packet by 0, the status that a node does not reply any packet by -1. Then we can get a list of status bits for the nodes on the routing path after the sink receives all the response packets from them (within a limited time cycle). The status for one round of response can be denoted by a vector $[b_1, b_2, \ldots, b_n]$, $\forall b_i \in \{-1, 0, 1\}$. After one round of response we get a vector of status bits. The sink can perform intrusion detection by analyzing the *status vector*.

To a status vector, $B = [b_1, b_2, \ldots, b_n]$, the sink finds all the status bits with value of -1 and adds the corresponding nodes to a set $\overline{S}_w$. $\overline{S}_w$ contains a collection of nodes that are without response to the sink. The nodes in $\overline{S}_w$ are considered as suspicious nodes rather than malicious nodes. It makes sense that some of the nodes on the routing path fail to receive or send packets because of interference or low communication quality. $\overline{S}_w$ is called a *suspicious set*. The nodes in the suspicious set are not excluded from the routing path. However, the sink pays more attention to these nodes in subsequent data collection. Then the sink tries to analyze the rest of B. To any $b_{i-1}, b_i \in B$, if b_{i-1}=0 or -1 and b_i=1, then b_{i-1} is a *change point* in B. A change point is a sensor node on the routing path where the value of status bit turns from 0 or -1 to 1.

Theorem: If S_c is a change point and S_{cd} is the nearest downstream node on the routing path, then the sequence (S_c, S_{cd}) contains a malicious node.

Proof: Without loss of generality we assume S_m is the nearest malicious node on the routing path to the sink. The nearest upstream node of S_m is denoted by S_{mu} and the nearest downstream node is denoted by S_{md}. We consider three specific nodes in the alert/response stage, the last node S_n on the routing path (from the source to the sink), which sends back a positive report packet, the nearest upstream node S_{nu} of S_n and the nearest downstream node S_{nd} of S_n. The sink only focuses on the response packets from {S_n, S_{nu}, S_{nd}}. We consider the three cases separately:

(1) S_m raises an alert falsely by sending back a negative report packet in order to disguise itself and deceive the sink. In this case S_n is S_{mu}, S_{nu} is the nearest upstream node of S_{mu} and S_{nd} is S_m.

(2) S_m does not raise an alert and sends back a positive report packet instead. In this case S_n is S_m, S_{nu} is S_{mu} and S_{nd} is S_{md}.

(2) S_m does not send back any response. In this case S_n is S_m, S_{nu} is S_{mu} and S_{nd} is S_{md}.

We can see that S_m falls into a sequence of nodes (S_{nu}, S_n, S_{nd}) in each case. Although it is difficult to determine which node in the collection is the malicious one, we can just set all the nodes in the collection as abnormal nodes and exclude them from the routing path. It is not necessary to distinguish between malicious nodes and threatened nodes close to the malicious nodes. It makes sense that the threatened nodes are not secure for routing. Both malicious and threatened nodes should be excluded from the routing path. Some existing works such as [5] figure out similar precautions in response to malicious nodes and threatened nodes. The sequence (S_{nu}, S_n, S_{nd}) is called the *malicious sequence*. To each case mentioned above the malicious node S_m always falls into a sub-sequence of the malicious sequence, which is (S_n, S_{nd}).

Therefore we can further reduce the malicious sequence to 2 nodes (S_n, S_{nd}). We can always get a malicious sequence with length of 2 by the intrusion detection algorithm. In Case 1 S_{mu} is a change point, while in Cases 2 and 3, S_m is a change point. The smallest malicious sequence always contains a change point as well as the nearest downstream node of the change point. We can always find a smallest malicious sequence by detecting the change point as well as the nearest downstream node, which contains a malicious node. The major goal of the intrusion detection algorithm is to find those smallest malicious sequences on the routing path. If there is more than one malicious node on the routing path, we should perform the above analysis for several rounds to get a series of malicious sequences.

Therefore we can get a final set of malicious nodes $\overline{S}_m = \bigcup_{i=1}^{k} \{S_n, S_{nd}\}_i$ by using the intrusion detection algorithm. The set $\overline{S}_m$ is called a *malicious set*. If the sink detects any two response packets that are contradictory, it just considers both of their senders as malicious nodes and adds them to $\overline{S}_m$. The sink can broadcast the identities of the nodes in $\overline{S}_m$ to the nodes in the WSN to exclude them from routing.

4 Conclusion

In this paper a novel secure mechanism called SERCID is proposed for WSNs. We have shown that the proposed approach is able to work even when some sensor nodes are compromised and become malicious nodes. The approach is able to protect the data collection in WSNs with malicious nodes. By comparison with existing research efforts the proposed approach is easily implemented and performed in resource-constrained wireless sensor networks. We provide a relatively simple but reliable approach to support secure data collection in wireless sensor networks. The work reported in this paper takes a step towards secure WSN.

Acknowledgements

This work is partially supported by a grant from NSFC Programs (NO. NSFC61003309, and NSFC60803161), an Educational Commission of Zhejiang Province Program (NO. Y200908082) and a Science and Technology Department of Zhejiang Province Program (NO. 2010C33045).

References

[1] Y. Zhang and W. Lee, Intrusion detection in wireless ad-hoc networks, *Proc. the 6th Annual International Conference on Mobile Computing and Networking*, 2000, 275-283.

[2] A. D. Wood and J. A. Stankovic. Denial of service in sensor networks. *IEEE Computer*, 2002, 35(10), 54-62.

[3] Y. Wang, G. Attebury and B. Ramamurthy, A survey of security issues in wireless sensor networks. *IEEE Commun. Surveys Tutorials*, 2006, 8(2), 2-23.

[4] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. Tygar, Spins: security protocols for sensor networks, *Proc. the 7th Annual International Conference on Mobile Computing and Networking*, 2001, 189-199.

[5] J. Deng, R. Han and S. Mishra, INSENS: Intrusion-tolerant routing in wireless sensor networks. *Computer Communications in Dependable Wireless Sensor Networks*, 2006, 29(2), 216-230.

Yuxin Mao received the PhD degree in Computer Science from Zhejiang University, China, in 2008. He is currently an Assistant Professor in the School of Computer and Information Engineering, Zhejiang Gongshang University. His research interests include Wireless Sensor Network, Semantic Web and Ontology Engineering.