A Feedback-based Multipath Approach for Secure Data Collection in Wireless Sensor Networks

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

ABSTRACT

In this paper, we propose a novel approach of secure data collection for wireless sensor networks. We explore secret sharing and multipath routing to achieve secure data collection in a wireless sensor network with compromised nodes. We present the notion of secure path, which makes full use of the routing functionality of wireless sensor networks, to support secure data collection. Moreover, we propose to perform intrusion detection for wireless sensor network based on the results of data collection.We evaluate the approach with a simulation experiment and analyze the simulation results in detail. We argue that the approach is simple but efficient to support secure data collection in wireless sensor network.

Keywords: Data Collection, Feedback, Intrusion Detection, Multipath, Security, Wireless Sensor Network.

1 INTRODUCTION

As a kind of wireless technology, wireless sensor networks (WSNs) are systems that comprise large numbers (usually hundreds or thousands) of wirelessly connected heterogeneous sensor nodes that are spatially distributed across a large field of interest [1]. However, the nature of WSNs makes them very vulnerable to an adversary's malicious attacks. An adversary can physically compromise a subset of sensor nodes in a WSN to eavesdrop information. The compromised nodes (or malicious nodes) become black holes in the WSN. Due to the unattended nature of WSNs, adversaries can easily produce such black holes [2]. Therefore, network security is an important issue to WSNs. Generally speaking, network security techniques are grouped into two categories: prevention-based techniques and detection-based techniques. When an intrusion takes place, prevention-based techniques are often the first line of defense against attacks, while detection-based techniques aim at identifying and excluding the attacker after the fail of prevention-based techniques. Detection-based techniques can be divided into two categories: misuse detection and anomaly detection. Misuse detection techniques match patterns of well-known attack profiles with the current changes, whereas anomaly detection uses established normal profiles and detects unusual deviations from the normal behavior as anomalies [3]. An attempt to apply the idea in WSNs makes a lot of sense to attacks.

Although intrusion detection is an important aspect to WSN, it is still in its infancy and there are currently only a few studies in this area. Due to some intrinsic features of WSN, it's difficult to perform efficient intrusion detection in a resource-restricted environment. Many intelligent or statistical approaches are too complex to WSNs. It's much easier to elude or bypass the holes rather than detect them. One possible solution to such kind of attack is to exploit the routing functionality of WSN. Specifically, if the locations of the black holes formed by the compromised nodes are known a priori, then sensed information can be delivered over paths that circumvent (bypass) these holes, whenever possible. As the existing intrusion detection methods for WSN are still immature, it's difficult to acquire such location information precisely in practice. Therefore the above idea of delivering information is often implemented in a probabilistic manner. Multipath routing allows the establishment of multiple paths between a single source and single destination node. It is typically proposed in order to increase the reliability of data transmission (i.e., fault tolerance) or to provide load balancing [4]. If the location information of compromised nodes is not known a priori, the source node can deliver sensed information by multiple paths to decrease the chance of the information being intercepted.

However, there are still problems with multipath routing approach. If the adversary can selectively compromise nodes, the sensed information is intercepted in each fixed routing path even if it can be distributed over different routes. One possible solution to this problem is delivering information randomly through different paths rather than fixed set of routes [5]. Although the adversary can still intercept part of information, we can reduce the probability of interception to an acceptable extent

by some mechanism.

In this paper, we propose a novel approach of secure data collection for WSN. We explore secret sharing and multipath routing to achieve secure data collection in a WSN with compromised nodes. Moreover, we propose to perform intrusion detection based on the results of feedback data collection for WSN. Compared with existing works [5][6][7][8]in this field, our approach use a novel tracing-feedback mechanism, which makes full use of the routing functionality of WSN, to improve the quality of data collection. The algorithms are easy to be implemented and performed in WSN.

2 MULTIPATH DATA COLLECTION

Generally speaking, a WSN [9][10] is a network composed of a large number of sensor nodes that are equipped with environmental sensors for temperature, humidity, pH value, etc. and can communicate with each other through a wireless radio device. A WSN is usually composed by two types of nodes: sensor nodes, and sink nodes. The sensor nodes, also known as motes or simply nodes, are small and constrained devices that have the ability of sensing the surrounding environment. The sink, also known as base station, is a more powerful node that behaves as an interface between the sensor nodes and the clients of the network. Sensor nodes in WSN are densely deployed either inside the phenomenon or very close to it. Although WSNs belong to the general family of wireless ad hoc networks, they have several distinctive features of their own [11]. For example, sensor nodes in WSN are small and inexpensive devices with restricted transmit power and energy supplies, compared with those in wireless ad hoc network. Multipath routing has been used for different goals in WSN, such as load balance, energy efficiency, etc. In this paper, we make use of multipath routing for secure data collection.

We use a (t, n)-threshold secret sharing algorithm, e.g., the Shamir's algorithm [12], to encode a packet of sensor data. When a sensor node wants to send a packet to the destination node (often the sink), it first breaks the packet into N shares according to the secret sharing algorithm. Each share is then transmitted to some randomly-picked neighbors. Therefore, we can break a data packet into a collection of shares using the (t, n)-threshold secret sharing algorithm and deliver different shares via different routing path (see figure 1).

We extend the algorithm given in [5] to randomly generate routing path for data collection. A data packet is broken into shares according to the (t, n)-threshold secret sharing algorithm and shares are transmitted to the sink via different paths. The algorithm in [5] does not consider the density of the sensor nodes in a WSN. If the degree or the number of neighbors of a node is small, there may be not enough candidates for delivering shares. Moreover, different nodes in a WSN have different degrees, a fixed (t, n)-threshold cannot satisfy every node in the WSN. We extend the algorithm in [5] with an adaptive (t, n)-threshold that varies according to the degree of node.

The adaptive data collection (ADC) algorithm is illustrated as follows:

(1) To a source node S that intends to send a data packet D, if its degree is larger than a threshold value k, set n to the degree of the node, which is the number of the neighbors of the node. Moreover, set t to a number that is less than n. Otherwise, the node sends D by normal routing until D reaches a node with enough degree.

(2) Break D into n shares according to the (t, n)-threshold secret sharing algorithm.

(3) To each share, perform a node selection using one of the four distributed random propagation mechanisms in [5].

(4) In this way, the shares of D are forwarded by a collection of relay nodes until they reach the sink.

In the algorithm above, we consider the degree of sensor nodes. If the degree of a sensor node is small, it's not necessary to break a data packet into shares. We use an adaptive mechanism to control process of breaking a data packet into shares. We will forward a data packet until it reaches a node that has enough degree for (t, n)-threshold algorithm. In order to distinguish data shares and the original data packet, we should add an additional flag in the beginning of the data frame.

3 FEEDBACK-BASED DATA COLLECTION ALGORITHM

The process of data collection in WSN is a relay of data packet from the source node to the sink. If the packet successfully arrives at the sink in the end, it means that there are no (or few) compromised nodes along the path. Therefore, we can make use of such historical information about data collection to improve the quality of the data collection and even perform intrusion detection.

3.1 Algorithm Description

We try to use a *tracing/feedback mechanism* for secure data collection. Therefore, we propose a feedback-based secure path (FSP) algorithm for this purpose. The algorithm is illustrated as follows:

(1) A source node S sends a data packet according to the ADC algorithm. To each share of data packet, S attaches an identity list L to it. Initially, L is an empty list.

(2) When a sensor node S_k receives a share, if it is a normal node, it adds its identity d_k to L.

(3) On the arrival of the share, the sink extracts $L=\{d_1, d_2, \ldots, d_n\}$ (d_i refers to the identity of

the node S_i) from the share and stores the pair <*S*, *L*> in its local database.

(4) The sink adds *L* to a notification packet and sends the packet to *S* according to *L*.

(5) When a sensor node S_j receives the packet, if its identity d_j is in *L*, it extracts a sub-path $P_j=\{d_{j+1}, d_{j+2}, \ldots, d_n\}$ from *L* and stores it into its local cache. S_j extracts its next-hop node S_{j-1} with identity d_{j-1} from *L* and forwards the packet to it.

(6) On the arrival of the packet, *S* extracts *L* from the packet, and stores it in its local cache. *S* also attaches a counter with an initial value λ to *L*. Here *L* is called a *secure path* for *S*. P_j is called a *secure path* for S_j (see figure 1).

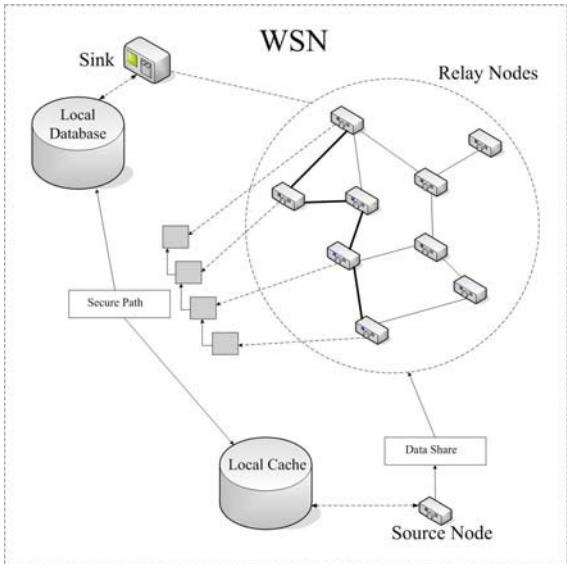

Figure 1: An illustration of the FSP algorithm for secure data collection.

In this algorithm, each normal sensor node in a routing path adds its unique identity to the data packet. When the data packet reaches the sink, it involves a routing path that consists of a list of the identities of normal sensor nodes. It means that the path is potentially secure for data collection and can be used again by the source node in the future. A complete secure path is always terminated and collected by the sink. Here we use a *feedback mechanism* to notify the source node that requires the path for future data collection. The sink sends back a notification packet that contains a secure path to the source node. The task of notification may be performed at intervals rather than immediately in order to reduce the overall consumption of the WSN.

Here the notion of secure path does not mean that the path is absolutely secure for data collection. A secure path may include compromised nodes on itself. It's mainly because that a compromised node drops a data packet with a probability. If a compromised node does not drop any data packet during the process of secure path construction, it will be considered potentially safe and be included in the path. In the worst case, each compromised node does not drop the data packet on the stage of secure path construction, in order to be involved in a secure path. Then each compromised node will appear on a secure path, which leads to a very low success ratio of data transmission. Therefore, when we say a path is a secure path, it only means that the path is currently safe for data collection.

3.2 Secure-Path based Data Collection

As long as a source node receives enough secure paths from the sink, it is able to send data via these paths. Therefore we can improve the ADC algorithm in section 2 by using secure paths. The secure-path based data collection (SPDC) algorithm is illustrated as follows:

(1) When a source node *S* intends to send a data share to the sink, it first checks its local cache. If there are secure paths, it selects a secure path $L=\{d_1, d_2, \ldots, d_n\}$ with the largest counter value from its local data repository. *S* adds *L* to the beginning of the data share. If there are no secure paths in the local cache of the relay node, it just performs random multipath routing as the ADC algorithm in section 2. If *S* has no secure paths at all, it performs path construction by using the FSP algorithm.

(2) Before sending the share, *S* first checks whether the node S_1 with identity d_1 is in its neighbor list. If the node is not the list, it just performs random multipath routing. Otherwise, it sends the share to the node S_1 for d_1.

(3) When a sensor node S_k receives a share, it first checks whether there is any secure path in the beginning of the share. If not, it performs random multipath routing and path construction. Otherwise, it checks whether the node S_{k+1} with identity d_{k+1} is in its neighbor list. If the node is not the list, it just performs random multipath routing. Otherwise, it sends the share to the node S_{k+1} for d_{k+1}.

If the share reaches the sink successfully:

(4) On the arrival of the share, if there is a secure path in the share, it means every relay node has used the path and the sink just sends back an empty notification to *S*. Otherwise, the sink extracts the identity list as a new secure path from the share, updates its local database and sends back a notification with the newly-constructed secure path to *S*.

(5) The relay nodes on the path update their local cache with secure path.

(6) On the arrival of the notification, *S* extracts the new secure path from the packet, and stores it in its local cache.

If the share is dropped or does not reaches the sink within the time span allowed:

(7) *S* does not receive a notification from the sink, and then it just decreases the counter of *P* by 1.

(8) If the counter of a secure path is cleared, *S* will remove it from its local cache.

From this algorithm, we can see that a secure path is not considered to be secure all the time. Secure paths are evaluated by their quality of service (QoS) for data collection. The SPDC algorithm deals with the problem of selective forwarding by using a scoring mechanism. We can exclude compromised nodes from data collection as many as possible. The local cache for secure paths of a source node therefore changes dynamically.

4 INTRUSION DETECTION BASED ON DATA COLLECTION

When the sink collects enough secure paths, it is able to find out compromised nodes by analyzing the secure paths. The sensor nodes appear in few secure paths are more likely to be compromised nodes. Therefore, we can make use of secure paths to perform intrusion detection. We propose an intrusion detection algorithm for WSN based on the results of FPDC. We argue that the algorithm is useful for detecting inside attacks like selective forwarding. The secure-path based intrusion detection (SPID) algorithm is illustrated as follows:

(1) To each sensor node, we assign it an initial reputation value γ.

(2) Assume the neighbor set of a sensor node *S* is $\{d_1, d_2, \ldots, d_n\}$. For a period of time, the sink counts the number of secure paths for each neighbor. The relative reputation for *S* is computed according to the following equation:

$$\tau = \frac{\sum_{i=1}^{n} \Delta k_i / \Delta K_i}{n} \quad (1)$$

where ΔK_i is the number of newly-generated secure paths the neighbor with identity d_i owns in all, and Δk_i is the number of newly-generated secure paths that involves *S* among those paths. If $\Delta K_i = 0$, we consider the value of $\Delta k_i / \Delta K_i$ as 0.

(3) The reputation increment of *S* is calculated according to the following equation:

$$E = e^{-N/\tau} \quad (2)$$

where *N* is the coefficient and we can set the value of *N* to 10.

(4) The reputation value of *S* is updated in the i^{th} round according to the following equation:

$$\gamma_i = (1-\alpha)* \gamma_i + \alpha * E_i, \gamma_0 = \gamma \quad (3)$$

Here α is used in as a reasonable balance between memory and current experience. We often set the value of α to 0.2. E_i is the reputation increment for *S* in the i^{th} round.

(5) If the reputation value of a sensor node is lower than a threshold value, we consider the node as a compromised node.

(6) The sink broadcasts the list of compromised nodes to the whole network. Source nodes can exclude these nodes in routing.

As long as the sink collects enough secure paths, it is able to perform the intrusion detection. In this way, it is no needs to collect additional information from sensor nodes to perform intrusion detection. It only makes use of the results of routing in WSN to support intrusion detection. The secure paths are constructed as a by-product of routing, and it places little burden on WSN with restricted computation and communication resources. Therefore, the algorithm is easy to be implemented in resource-constrained WSN.

5 SIMULATION AND EVALUATION

In this section, we construct simulation to evaluate the performance of the proposed approach. The major metric for performance evaluation is the *packet interception probability* (PIP) for a source node, defined as the ratio of the number of intercepted data packets to the total number of packets sent from the source node. To better understand the capability of these randomized multi-path routing algorithms in bypassing black holes, we also compare the performance of our approach with the original algorithms in [5].

The basic setting for the simulation is illustrated in table 1. Here the parameter *drop rate* refers to the probability that a compromised will drop a data packet.

Table 1: The major parameters for the simulation.

Parameter	Value
Sensor Node Number	50
Drop Rate	0.2
Threshold Value *k*	5
Initial Counter Value λ	3
Number of Source Nodes	10

5.1 Packet Interception Probability Evaluation

We first fix the location of the source node that sends data the sink. We first investigate the PIP for the source node under different numbers of compromised nodes. For each number of compromised nodes, we evaluate the average PIP for the source node. Figure 2 shows a plot of the PIP for the source node under different numbers of

compromised nodes. It's obvious to see that the PIP increases when the number of compromised nodes becomes larger. When half of the sensor nodes are compromised nodes, most of the data packets are intercepted. We also compare the performance of the SPDC algorithm with that of NRRP algorithm proposed in [5]. As can be seen in the figure, the performance of SPDC is better than NRRP with the same number of compromised nodes. When the number of compromised nodes is small or large, the performance of the two algorithms is very close. However, SPDC behaves much better than NRRP with number of compromised nodes falling into the extent (13, 18).

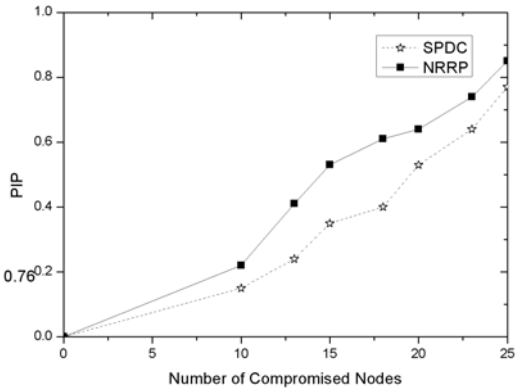

Figure 2: The PIP values for SPDC and NRRP with different numbers of compromised nodes.

5.2 Performance Evaluation with Source Node Set

The work in [5] performs simulation by using a fixed source node that sends data to the sink. Using only fixed source node is not enough to simulate the behaviors of WSN. In practice, data are always generated by different sensor node distributed across an area. In fact, our approach can achieve better performance as long as there is secure path from the source node to the sink. It is insufficient to evaluate the PIP with a fixed source node. Therefore we evaluate the overall PIP with a collection of source nodes. We select a collection of source nodes and each one is likely to generate data and send it to the sink. Then we evaluate the overall performance for our approach for the collection of source nodes.

The process of simulation with a collection of source nodes is similar with that of a fixed source node. We run the data collection algorithms for each source node in the collection and record the accumulative result for the collection. Here the number of source nodes in the collection is 10 (see table 1). As can be seen in figure 3, the performance of SPDC is better than NRRP with the same number of compromised nodes.

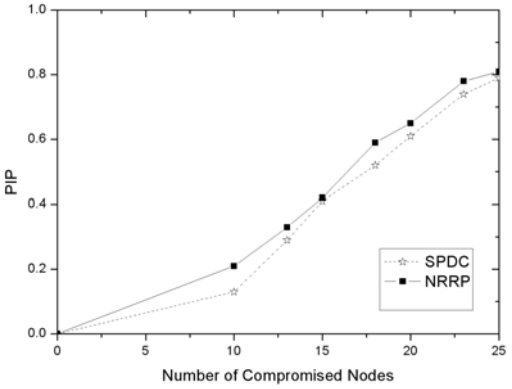

Figure 3: The PIP values for SPDC and NRRP with a collection of source nodes.

6 RELATED WORKS

There have been a few of on-going efforts about multipath routing for secure data collection presented in literature. For example, the SPREAD algorithm in [10] attempts to find multiple most-secure and node-disjoint paths. A modified Dijkstra algorithm is used to iteratively find the top-K most secure node-disjoint paths. The H-SPREAD algorithm [11] improves the SPREAD algorithm by simultaneously accounting for both security and reliability requirements. The work in [12] presents distributed Bound-Control and Lex-Control algorithms, which compute multiple paths respectively. Shu et al. in [5] present an approach for secure data collection by using (t, n)-threshold secret sharing algorithm and randomized multipath routes. A packet is broken into shares, which are sent to the sink through randomly-generated paths. Compared with our approach, they use a fixed source node to evaluate the approach in simulation, while we extend their simulation with a collection of source nodes. Nasser and Chen in [13] propose a routing protocol that uses multipath alternately as the path for communicating between two nodes. The protocol defends against some specific attacks like selective forwarding by advertising an attractive route to the destination. Deng et al. in [14] propose an intrusion-tolerant routing protocol for WSNs. They try to preserve WSN security by using one way hash chains and nested keyed message authentication codes, as well as multipath routing.

7 CONCLUSION

In this paper, we propose a novel approach of secure data collection for WSN. We explore secret sharing and multipath routing to achieve secure data collection in a WSN with compromised nodes. We propose to use a novel tracing-feedback mechanism, which makes full use of the routing functionality of WSN, to improve the quality of data collection. Moreover, we propose to perform intrusion detection for wireless sensor network based on the results of

data collection. Moreover, we propose to perform intrusion detection for wireless sensor network based on the results of data collection. Compared with existing works in this field, our approach use a novel tracing-feedback mechanism, which makes full use of the routing functionality of WSN, to improve the quality of data collection. The major difference between our approach and the existing multipath methods is that the secure paths here are potentially safe for data collection. In all, our work tries to take a step forward secure data collection for WSN.

Future works may include: (1) improve the efficiency of the algorithms; (2) evaluate the intrusion detection algorithm under different conditions; (3) considering a more complex WSN model to evaluate the algorithm.

ACKNOWLEDGEMENT

This work is partially supported by a grant from a NSFC Program (NO. NSFC60803161), a Science and Technology Program of ZJGSU (NO. 1130XJ200920), and also a grant from Educational Commission of Zhejiang Province (NO. Y200908082).

8 REFERENCES

[1] K. S. Low, W. N. Win, and M. J. Er: Wireless Sensor Networks for Industrial Environments, Mater. Sci. Forum, Vol. 119, pp. 83-87 (1992).

[2] F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci: A survey on sensor networks, IEEE Communications Magazine, Vol. 40(8), pp. 102–114 (2002).

[3] Y. Zhang, and W. Lee: Intrusion Detection in Wireless Ad-Hoc Networks, Proc. the 6th Annual International Conference on Mobile Computing and Networking, (2000).

[4] A. Tsirigos, and Z.J. Haas: Multipath routing in the presence of frequent topological changes, IEEE Communication Magazine, Vol. 39(11), pp. 132-138 (2001).

[5] T. Shu, S. Liu, and M. KrunzSecure: Data Collection in Wireless Sensor Networks Using Randomized Dispersive Routes, Proc. IEEE INFOCOM Conference, pp. 2846-2850 (2009).

[6] W. Lou, and Y. Kwon: H-spread: a hybrid multipath scheme for secure and reliable data collection in wireless sensor networks, IEEE Transactions on Vehicular Technology, Vol. 55(4), pp. 1320–1330 (2006).

[7] N. Nasser, and Y. Chen: SEEM: Secure and energy-efficient multipath routing protocol for wireless sensor networks, Computer Communications, Vol. 30(11-12), pp. 2401-2412 (2007).

[8] J. Deng, R. Han, and S. Mishra: INSENS: Intrusion-tolerant routing for wireless sensor networks, Computer Communication, Vol. 29(2), pp. 216-230 (2006).

[9] F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci: Wireless Sensor Networks: A Survey, Computer Networks, Vol. 38, pp. 393-422 (2002).

[10] K. Ioannis, T. Dimitriou, and F.C. Freiling: Towards Intrusion Detection in Wireless Sensor Networks, Proc. the 13th European Wireless Conference, (2007).

[11] I. Onat, and A. Miri: An Intrusion Detection System for Wireless Sensor Networks, Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (2005).

[12] A. Shamir: How to Share a Secret, Communication of the ACM, Vol. 22(11), pp. 612-613 (1979).