International Journal of Latest Engineering Research and Applications (IJLERA) ISSN: 2455-7137

Volume – 02, Issue – 04, April – 2017, PP – 61-66

An Improved Link Failure and Malevolent Data Detection Method for Secure Data Transmission in Wireless Sensor Network

Vinay shrivastava
Student, Department of Computer Science and Engg
Shri Balaji Institute of Technology and Management, Betul (MP),

Ravi H Gedam
Assistant Professor, Department of Computer Science and Engg.,
Shri Balaji Institute of Technology and Management, Betul (MP)

Abstract: WSNs have received noteworthy consideration in current years due to their potential applications in wildlife tracking, armed sensing, traffic investigation, fitness care, atmosphere monitoring, building constructions monitoring, etc. Nodes in WSNs are disposed to letdown due to hardware letdown, energy reduction, communication link faults, mischievous attack, and so on. Every mobile node is free to move in any ways and can change their link at any time. Error free and reliable data transfer between source and destination is the challenges in WSN. Damaged link discovery plays an important part in network failure detection and network management. Malicious data injection plays a noteworthy contribution in network disaster detection and network management. Malicious data injection may cause failure of link in network. The proposed system will enthusiastically detect the link failure and malicious data injection in wireless sensor network. This paper proposes a method which automatically discover link failure detection and malicious data injection in node which may cause network failure. After detection of link failure and malicious data injection data can be securely transferred to the destination and improve the performance of wireless sensor network.

Keywords: Wireless Sensor Network, Security, link failure, malicious data injection.

I. Introduction

Associated to the wired networks, it seems considerable more important to sense malicious data injection rather than node responsibilities in WSNs[1]. A wireless sensor network comprises of several small sized sensor nodes that have computation capabilities. Wireless sensor networks (WSNs) have been widely used in many application areas such as infrastructure protection, environment monitoring and habitat tracing[2]. Error free and reliable data transfer between source and destination is the challenges in WSN. Malicious data injection[1] may cause failure of link in wireless sensor network. Our technique enthusiastically detect the malicious data injection and immediately report to the system and perform data transmission with secure route. Here verification framework is used to remove outside competitors and guarantee that only permissible nodes accomplish certain operations. The objective of our work is to detect malicious data injection in WSN, to determine route[3] for protected data transmission. Malicious data injection plays a noteworthy role in network failure detection and network administration. Our technique enthusiastically detect the malicious data injection and immediately report to the system and perform data transmission with secure route. With the help of reverse route the node will send malicious data injection message to the upstream node. This error message is used by node to detect malicious data injection in the wireless sensor network. The backup node is used to secure data transmission.

Damaged link discovery[4] plays an important part in network failure detection and network management. After detection of damaged link data can be securely transferred to the destination and improve the performance of wireless sensor network. The data transmission is possible only if like failure does not occur in wireless sensor network. In any circumstances if link down the network cannot continue to transfer the data to destination. With the help of reverse route the node will send link damage message to the upstream node. This error message is used by node to detect link failure in the wireless sensor network. This error message is used by node to change the protected and backup route for better data transmission. The backup route cache[5] is fetched from the backup node to check link damage failure. This message is used by backup node to replace the contents of data packet. This packet is used to inform all the nodes about route changes in the network. After getting the message source node directs the packets with new and secured node.

The rest of the paper is organized as follows.
Section 2 provides the background, relevant for the context. Section 3 provides the proposed methodology, proposed algorithm and description of proposed methodology. Section 4 represents the implementation of proposed methodology, discussion on simulation Results and performance analysis of simulation results. Section 5 concludes the paper with a summary of the main findings concluding remarks, limitation discussion and an outlook on future research directions.

II. Literature Survey

A wireless sensor network comprises of several small sized sensor nodes that have computation capabilities. Wireless sensor networks (WSNs) have been widely used in many application areas such as infrastructure protection, environment monitoring and habitat tracing. Associated to the wired networks, it seems considerable more important to sense malicious data injection rather than node responsibilities in WSNs.
[5]method propose a novel algorithm to identify malicious data injections and build measurement estimates that are resistant to several compromised sensors even when they collude in the attack. The approach pursued in this paper is based on measurements analysis and its applicability relies on the assumption that the measurements are correlated under genuine circumstances, while compromised measurements disrupt such correlations. The demerits of the system is that the measurements contain redundant information. This will not detect unpredictable changes in the spatial patterns. [6] system will work where the value of the sensed phenomenon is required to be the same within a neighbourhood and measurements differ only because of noise. This assumption allows to easily estimate the ground truth and label the measurements as outlying through. However, this assumption is generally valid only for very small neighbourhoods, where collusion attacks can be successful by compromising all the sensors. The authors [7] have proposed to detect inconsistencies in the correlation within a neighbourhood by extracting a unique overall consistencymetric, to which every neighbor contributes. This method may disrupting the reported values of the WSN. The estimates is aggregated with a collusion-resistant operator that produces a final reliable estimate to be compared with the reported measurement. An approach[8] similarly based on aggregation of individual sensors' information is majority voting where each sensor votes for a neighbor's maliciousness and the votes are aggregated by majority. Similarly, trust-management frameworks aggregate individual beliefs about a sensor's behavior. A sensor's behavior is mapped to a trust value by all its neighbors, and then the sensor's trustworthiness is obtained e.g., by averaging the trust values. The main drawback of these techniques[9] is that they introduce an additional variable—the vote, or trust value—about which an attacker can lie with or without lying about the measurements at the same time.
Low-energy adaptive clustering hierarchy (LEACH): LEACH [36,37] is the first and most popular energy-efficient hierarchical clustering algorithm for WSNs that was proposed for reducing power consumption. The operation of LEACH is divided into rounds having two phases each namely (i) a setup phase to organize the network into clusters, CH advertisement, and transmission schedule creation and (ii) a steady-state phase for data aggregation, compression, and transmission to the sink.

LEACH is completely distributed and requires no global knowledge of network. It reduces energy consumption by (a) minimizing the communication cost between sensors and their cluster heads and (b) turning off non-head nodes as much as possible [10]. LEACH uses single-hop routing where each node can transmit directly to the cluster-head and the sink. Therefore, it is not applicable to networks deployed in large regions. Furthermore, the idea of dynamic clustering brings extra overhead, e.g. head changes, advertisements etc., which may diminish the gain in energy consumption.

III. Proposed Work

In proposed work the packet delivery ratio is compared with the threshold value. If packet delivery ratio drop to the threshold value then Source node randomly choose the cooperative bait address, of one node neighbor to bait malicious node, then send bait request. If any node reply RREP from other route except neighbor node then start the reverse tracing program and send test packets, check messages to detect malicious node, source node list malicious node onto malicious data injection list, set alarm packet and end the transmission. The first step in our work is to setup the scenario i.e. to setup the node used in algorithm. To setup the source and destination used in the system. Set the threshold value for packet delivery ratio. It also set the routing parameters, routing protocols, packet size, dimensional area, and rate of transmission. The next step is to send the request generated by source RREQ. The next step is to check whether the source get the reply RREP by valid and authenticated node. Because the malicious data injection node can also generate the RREP signal. If message from the authenticated node then system is marked as an authenticated and source can transmit data to the specified and secured path. If RREP reply is from invalid or unauthenticated node then first count the number of hops. If number of hop counts exceeded then marked system is invalid and exit from the network. If number of nodes count is less than system may occur link failure report to the system. To find another secure neighbor node go to the RREQ source request step.

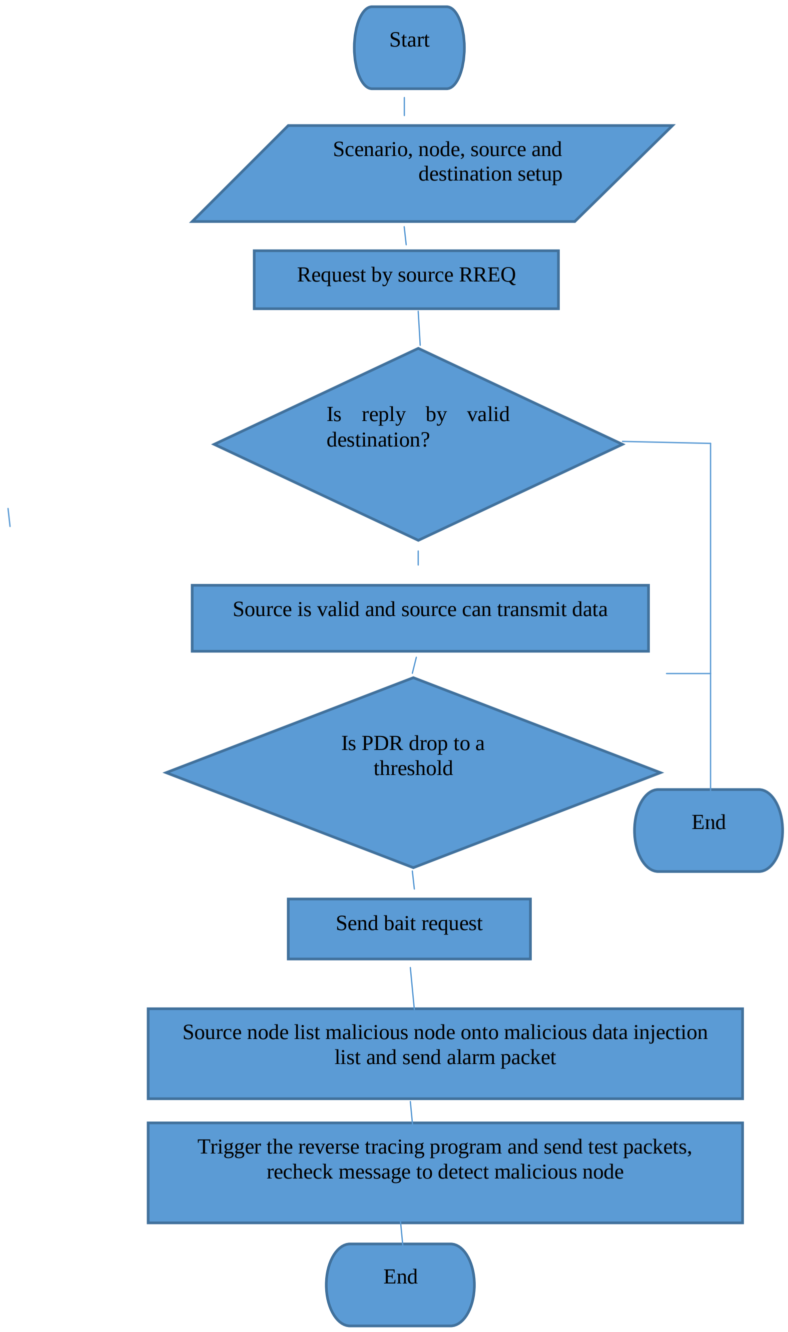

Start
Scenario, node, source and destination setup
Request by source RREQ
Is reply by valid destination?
Source is valid and source can transmit data
Is PDR drop to a threshold
End
Send bait request
Source node list malicious node onto malicious data injection list and send alarm packet
Trigger the reverse tracing program and send test packets, recheck message to detect malicious node
End

Fig 1 Flow diagram of proposed algorithm

```
Step 1: Scenario setup, Node setup, Routing protocol setup, Source and destination setup
                    Threshold value setup
Step 2: Request send by source RREQ
Step 3: Check whether reply RREP by valid and authenticated node
If node is authenticated then
    Marked system is valid
    Source can transmit data
Else if
```

```
Check hop count of the system
    If hop count exceeded then
        System is invalid
    Goto End
   Else if hop count is less then
        Linked failure in system
        Report to the system
   Else
                  Goto step 2 request send by RREQ
 End if
Step 4: Check packet delivery ratio of the system
    If packet delivery ratio drop to the threshold then
                  Source node randomly choose the cooperative bait address of one node neighbor to bait malicious node
        Send bait request
If any node reply RREP from other route except neighbor node then
     Start the reverse tracing program and send test packets
     Check messages to detect malicious node
     Source node list malicious node onto malicious data injection list
    Set alarm packet
                  Goto End
Else
                  Goto End
End if
End
```

The next step is to check packet delivery ratio of the network. The packet delivery ratio is compared with the threshold value. If packet delivery ratio drop to the threshold value then Source node randomly choose the cooperative bait address, of one node neighbor to bait malicious node, then send bait request. If any node reply RREP from other route except neighbor node then start the reverse tracing program and send test packets, check messages to detect malicious node, source node list malicious node onto malicious data injection list, set alarm packet and end the transmission.

IV. Implementation

We used NS2 simulator for implementation of proposed work. We also used C/C++ and TCL language for implementation. We performed our experiment in PIV 2.0 GHz machine with 2GB RAM. In our simulation work, we have different the amount of nodes from 50 to 100, which are arbitrarily positioned in dissimilar parts of positioning part with a static density. For this simulation, we have used the network parameters, such as Dimension, Number of nodes, traffic, transmission rate, Routing protocol, transmission range, sensitivity, transmission power etc., are used. We have performed our experiment with different number of nodes, with or without mobility. The dimensional area and speed of the scenario is also changed according to situation. With the help of reverse route the node will send link damage message to the upstream node. This error message is used by node to detect link failure in the wireless sensor network.

Table 1 Simulation scenario

Quantity of nodes	50, 100, 150, 200, 250, 300
Simulated area dimension	810×610
Routing Procedure	LEACH
Simulation time in seconds	110
Transport Layer	FTP, TCP
Traffic flow type	CBR
Packet size in bytes	1010
Quantity of traffic links	20 , 8
Max. Speeds in m/s	30

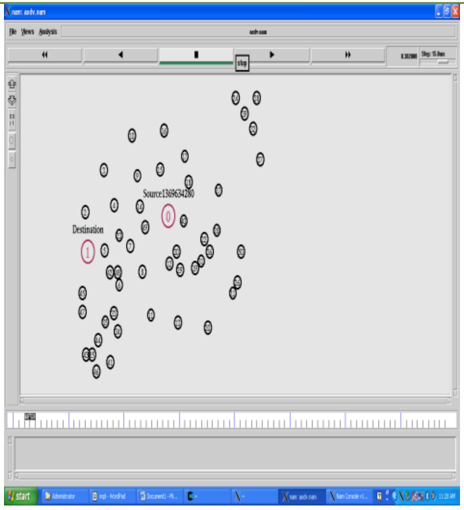
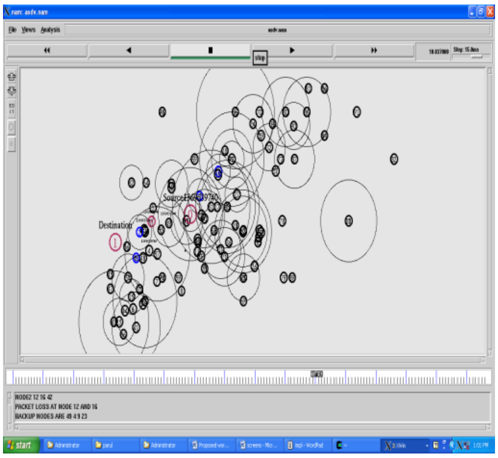


Figure 1 Simulation setup



```
NODE2 12 16 42
PACKET LOSS AT NODE 12 AND 16
BACKUP NODES ARE 48 4 9 23
```

Figure 2 Simulation result

Simulation readings of the proposed protocol are carried out to estimate its performance, and compared its performance. Fig. 2 represents the data transfer percentage of all the three routing protocols. It is noted that the data transmission ratio of all the set of rules rise as the node compactness increases. When node density is very high, there are additional nodes available for data promoting, and this rises the delivery ratio.

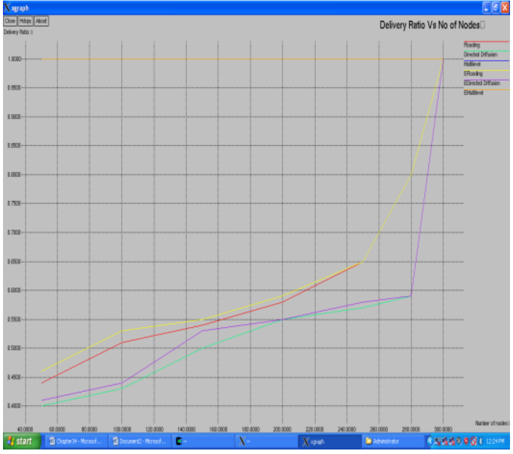


Figure 3 Performance Graph for Delivery Ratio Vs Number of Nodes

Fig. represents the data transfer percentage of all the three routing protocols. It is noted that the data transmission ratio of all the set of rules rise as the node compactness increases. When node density is very high, there are additional nodes available for data promoting, and this rises the delivery ratio. Overflowing offers less data delivery rates, followed by flooding is directed diffusion; it did not familiarize well its performance to network size growth. The multilevel routing protocol has preserved continuous transport rates throughout the simulated situations. This is an outcome of the influence of the process it uses to create a routing route.

V. Conclusion

A wireless sensor network (WSN) comprises of various very small sized sensor computational nodes that have computation control. Nodes in WSNs are disposed to letdown due to hardware letdown, energy reduction, communication link faults, mischievous attack, and so on. Malicious data injection may cause failure of link in wireless sensor network. A wireless link itself nearly exists, which means we can’t directly see and appraise whether it achieves well or not. Here verification framework is used to remove outside competitors and guarantee that only permissible nodes accomplish certain operations. The key is used for secure data transmission. Our technique enthusiastically detect the malicious data injection and immediately report to the system and perform data transmission with secure route. This paper proposes a structure which automatically discover malicious data injection in node which may cause link failure and discover secure shortest path for data transmission. After detection of malicious data injection data can be securely transferred to the destination and improve the performance of wireless sensor network.

References

[1]. Vittorio P. Illiano and Emil C. Lupu, Detecting Malicious Data Injections in Event Detection Wireless Sensor Networks, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 12, NO. 3, SEPTEMBER 2015, pp-496-512

[2]. T. S. Rappaport et al., Wireless Communications: Principles and Practice, vol. 207, Englewood Cliffs, NJ, USA: Prentice-Hall, 1996.

[3]. W. Dong, Y. Liu, Y. He, T. Zhu, and C. Chen, “Measurement and analysis on the packet delivery performance in a large-scale sensor network,” IEEE/ACM Trans. Netw., vol. 22, no. 6, pp. 1952–1963, Dec. 2014.

[4]. H. Chang et al., “Spinning beacons for precise indoor localization,” in Proc. ACM SenSys, Raleigh, NC, USA, 2008, pp. 127–140.

[5]. Qiang Ma, Kebin Liu, Zhichao Cao, Tong Zhu, Yunhao Liu, Link Scanner: Faulty Link Detection for Wireless Sensor Networks, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, VOL. 14, pp 4428-4438, Aug 2015

[6]. S. S. Ahuja, S. Ramasubramanian, and M. M. Krunz, “Single-link failure detection in all-optical networks using monitoring cycles and paths,” IEEE/ACM Trans. Netw., vol. 17, no. 4, pp. 1080–1093, Aug. 2009.

[7]. Q. Cao, T. Abdelzaher, J. Stankovic, K. Whitehouse, and L. Luo, “Declarative tracepoints: A programmable and application independent debugging system for wireless sensor networks,” in Proc. ACM SenSys, Raleigh, NC, USA, 2008, pp. 85–98.

[8]. A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin, “Statistical model of lossy links in wireless sensor networks,” in Proc. IEEE IPSN, 2005, pp. 81–88.

[9]. L. Girod et al., “EmStar: A software environment for developing and deploying wireless sensor networks,” in Proc. USENIX Annu. Tech. Conf., Boston, MA, USA, 2004, p. 24.

[10]. Y. Hamazumi, M. Koga, K. Kawai, H. Ichino, and K. Sato, “Optical path fault management in layered networks,” in Proc. IEEE GLOBECOM, Sydney, NSW, Australia, 1998, pp. 2309–2314.