NETWORK LAYER BASED SECURE ROUTING PROTOCOL FOR WIRELESS AD HOC SENSOR NETWORKS IN URBAN ENVIRONMENTS

Sanjay K. Dhurandher [1], Mohammad S. Obaidat [2], Deepank Gupta [1], Nidhi Gupta [1] and Anupriya Asthana[1]

[1]*Division of Information Technology, Netaji Subhas Institute of Technology, University of Delhi, New Delhi, India*
[2]*Department of Computer Science & Software Engineering, Monmouth University, NJ, U.S.A.*

Keywords: Routing protocols, Security, Wireless sensor networks.

Abstract: Security is essential in wireless sensor networks as they are being used in urban environments, life saving disaster management and rescue operations. Any serious attack at the routing layer can cause serious damages. Although a lot of security measures have been proposed for application and transport layer, we have found that there is not enough research geared towards securing the network at the routing layer. In this paper, we propose a novel solution for securing against external as well as internal attacks. The protocol maintains a working network by using redundant multiple paths despite attacks at one route. It also identifies and removes the malicious nodes from the system. Since the system is totally distributed and does not require a central server as required in some of the other protocols, there is no single point of failure. We also keep in mind the limited computing resources and network bandwidth of the wireless sensor nodes. Finally the paper quantifies the protocol's effectiveness against some of the existing secure routing protocols namely QDV and SNEP using simulation studies.

1 INTRODUCTION

Wireless Sensor Networks (WSNs) (Mainwaring et al., 2006) are among the popular emerging technologies that open a wide perspective for future applications in ubiquitous computing and ambient intelligence. Wireless sensor nodes form a dense, large scale network and are expected to function unattended. Their hardware limitations have led to the design of new protocols at the lower communication levels, such as physical, MAC (Demirkol et al., 2006) and routing. WSNs can be thought of as an extension to MANETs since special sensor nodes are included into the wireless network to give further utilities or increase the services given by it. Sensor nodes are stations which sense variables about the physical world around them. Generic nodes in the network sense network characteristics whereas sensor nodes sense changes in the environment in which they exist.

Ensuring security is critical in any WSN. If any sensor node is hacked or information about various environment variables fails to reach the desired controllers, the consequences can be dire. Although sensor networks share many characteristics with wireless ad hoc networks, one of the major differences is the energy and computational resources available at sensor nodes. Therefore, any security protocol for WSNs must keep in mind, the relatively constrained energy and computational resources in mind. With this criterion in mind, we find that many existing protocols fail to fit in WSNs.

Routing protocols proposed for WSNs cope well with the dynamically changing topology but many of them have no mechanisms to defend against malicious attacks. We are convinced that security problems cannot be considered separately and must be taken into account for the specification of all the functionalities of the network. Since, research on protocols is still going on, there is no single standard routing protocol. Therefore, we aim to capture the common security threats and try to provide a secure existing routing protocol in WSNs.

In most of the routing protocols, routers exchange information based on the topology of the network in

K. Dhurandher S., S. Obaidat M., Gupta D., Gupta N. and Asthana A. (2010).
NETWORK LAYER BASED SECURE ROUTING PROTOCOL FOR WIRELESS AD HOC SENSOR NETWORKS IN URBAN ENVIRONMENTS.
In *Proceedings of the International Conference on Wireless Information Networks and Systems*, pages 23-28
DOI: 10.5220/0003105100230028

order to establish routes between nodes. Such information could become a target for malicious adversaries who intend to bring the network down. There are two sources of threats to routing protocols. The first comes from external attackers. By injecting erroneous routing information, replaying old routing information, or distorting routing information, an attacker can successfully partition a network or introduce excessive traffic load into the network by causing retransmission and inefficient routing. There are two ways of ensuring security: either by encrypting the data or by identification and removal of malicious node from the network. The first approach using public-key algorithms such as the Diffie Hellman (Perrig et al., 2001) is not suitable for a wireless sensor network in which nodes have limited computing resources. Using such an algorithm will mean a large amount of computation power being used to encrypt and decrypt every message making the network slow.

The second and more severe kind of threat comes from the compromised nodes, which might advertise incorrect routing information to other nodes. Detection of such incorrect information is difficult. Merely requiring routing information to be signed by each node would not work, because compromised nodes are able to generate valid signatures using their private keys. To defend against the first kind of threats, nodes can protect routing information in the same way they protect data traffic, i.e., through the use of cryptographic schemes [3, 4] such as digital signature. However, this defense is ineffective against attacks from compromised servers. Worse yet, as we have argued, we cannot neglect the possibility of nodes being compromised in an ad hoc network. Detection of compromised nodes through routing information is also difficult in an ad hoc network because of its dynamically changing topology: when a piece of routing information is found invalid, the information could be generated by a compromised node, or, it could have become invalid as a result of topology changes. It is difficult to distinguish between the two cases. On the other hand, we can exploit certain properties of ad hoc networks to achieve secure routing. Note that routing protocols for ad hoc networks must handle outdated routing information to accommodate the dynamically changing topology. False routing information generated by compromised nodes could, to some extent, be considered outdated information. As long as there are sufficiently many non-malicious nodes, the routing protocol should be able to find routes that go around these compromised nodes. Such capability of the routing protocols usually relies on the inherent redundancies, multiple, possibly disjoint, routes between nodes in ad hoc networks.

With the above premise in mind; we propose a solution to work around malicious nodes and also to detect and remove the malicious nodes when found from the trusted population (other nodes to which a node transmits data). The solution to this has been proposed in this paper which is an extension of the proposed energy efficient protocol in (Sanjay et al., 2010). We have named this protocol as *Dynamic Energy Efficient and Secure Routing (DEESR)* protocol. This can easily be extended to other cluster based routing protocols. The distributed algorithm applied in this protocol does not require excessive computation resources. Also, there is no extra network cost involved with this protocol. The paper goes on to compare the proposed DEESR solution with some of the earlier proposed secure routing protocols using simulation techniques. The paper has been divided as follows. Section 2 discusses existing methods of achieving secure routing by discussing some of the well known secure routing protocols. Section 3 describes the terms and terminologies used throughout the paper. In section 4, we provide a detailed explanation of the proposed DEESR protocol. Simulation results comparing DEESR protocol with the existing approaches are presented in Section 5. Lastly, Section 6 summarizes our finding and provides insights into the future works.

2 PREVIOUS WORKS

There are various security based protocols (Perrig et al., 2001), (Sanjay et al., 2009), (Yi et al., 2001), (Karlof et al., 2004) for ad hoc networks. Yi (Yi et al., 2001) have discussed in their paper that if the routing protocol is compromised by changing the messages in the transit, then no security at higher layers can help. To address this problem they have proposed *Security Aware Ad-hoc Routing (SAR)*. It makes sure that data is routed through a secure route composed of trusted nodes and the security of the information in the routing protocol. Apart from this, security has been implemented at link layer in TinySec protocol given in (Karlof et al., 2004).

SPINS (Perrig et al., 2001) was proposed keeping in mind the resource limitations. This protocol encrypts a message differently each time. SPINS comprises of two building blocks, SNEP and µ-TESLA. SNEP protocol was designed for stationary networks and assumed that the base station is trusted. It also assumed an access point for the other nodes in the network. The µ-TESLA protocol is based on key

chain generation, in which a key is generated from the next key, in addition to the application of a function and the last key is generated once in a time-interval. But the encryption approach still requires storage of public and private keys. Also, the transmission of digital signatures consumes more energy. Though, it ensures data authentication and prevents other nodes from modifying and reinserting packets into the network, it does not prevent them from overhearing the data-packets.

Another security routing protocol, QDV (Sanjay et al., 2009) is based on quality of the route taken by the packets to be transmitted between nodes in the network. The route taken, by the packet, is governed by different parameters based on the quality of service and quality of security and also the future benefits in the transmission.

This protocol (QDV) designed for securing the wireless sensor networks and is based on Ant colony optimization (ACO) (Dorigo and Stuetzle, 2004). It uses quality-of- service (QoS) and reputation of the node to find out the trust of the neighbor node. Thus, by monitoring these two parameters the protocol is able to detect and disable the malicious nodes from gaining access and participating in the network. Though it takes into account the malicious activity of packet drop and reinsertion, it does not secure the data in the data packet. Also it is not energy efficient for a resource constrained environment.

3 TERMS AND NOTATIONS

The DEESR protocol has been designed so that a sensor ad hoc network can be set up for mobile devices securely and in an energy efficient manner. To facilitate this, certain parameters have been focused upon. These parameters are then used to reach to a routing decision. The following subsections present the terms and parameters used in the protocol as it is necessary to understand them before proceeding.

Population

Every node in the network has a set of nodes which are in its transmission range and with which it decides it can communicate securely. This set of nodes with which a node communicates with is called the population of that node. Only the nodes satisfying a certain criteria are included in the population of a node. In this protocol, as security is of prime importance, the parameter used for narrowing down a node's population is the Dynamic Trust Factor (DTF).

To limit the database requirements and to minimize network traffic for population maintenance, a node maintains a population restricted up to a maximum population size. The population size is dependent on the computing resources and network bandwidth of a node.

Dynamic Trust Factor (DTF)

This is given by the ratio of the sum of the packets received and generated subtracted by the number of packets transmitted to the sum of the packets received and generated. Thus, DTF gives the ratio of the number of packets dropped to the total number of packets received and generated. In DEESR protocol a particular node is trusted if its packet dropping ratio is less. Thus, trust is inversely proportional to the number of packets dropped. We do not consider or handle the situation where the packet is tampered with at the next-hop before retransmission. This value is also extracted from the routing table. Equation (1) shwon below represents how the DTF is calculated:

$$\delta = (p_r + p_g - p_d) / (p_r + p_g) \quad (1)$$

Where:

δ denotes the Dynamic Trust Factor (DTF)
Pr = number of packets received by that node
p_g = number of packets generated by that node
p_d = number of packets dropped by that node

Dynamic Trust Factor (DTF) is given by the ratio of the packets transmitted by a node to the total number of packets received and generated at that particular node. Subtracting the number of packets dropped (p_d) from the sum of the number of packets received (p_r) and packets generated (p_g) results in the number of packets transmitted. Similarly the sum of p_r and p_g results in the total number of packets at a particular node. At the source and the destination nodes, the packets for which the node is the source/destination are not counted in the DTF calculation, which provide security against active attack, specifically, malicious packet injection.

4 PROTOCOL AND EXPERIMENT

In this section, the functioning of the proposed protocol has been explained. There are four major parts of the protocol: route initiation, population maintenance, updating of next-hops and route-error respectively. The route initiation and data packet

transmission are not explained in major detail in this section. Most of the focus is on population maintenance and updating of next-hops.

Route Initiation

Let us consider a scenario in which a node *A* acts as a source and requires sending data to the destination node *E* according to the topology shown in Figure 1. *A* broadcasts the route-request packet. This packet is received by, say, node *B* which updates the information about *A* in its routing table and broadcasts the packet again. Similarly, this goes on till the packet reaches the destination node *E* with the information of the previous node. Since *E* is the destination node, it initiates a route-reply packet and sends it to all the previous nodes from which it has received the request since there maybe more than one route between the source and destination. The first hop in the reply route updates the information of the destination node and again replaces that information with its own information and unicasts the packet back to previous nodes in the route. When the reply packet finally reaches the source node, it makes an entry in its routing table for the routes.

In Figure 1, let us assume that a malicious node *F* injects packets into the network disguising itself as a trustworthy node, say, node *A*. Since *F* is not the actual source, so the disguised packet will be tracked down and the DTF will be calculated which will indicate its untrustworthiness. Hence, DEESR protocol will be able to isolate such a node from the network.

Attack during route initiation

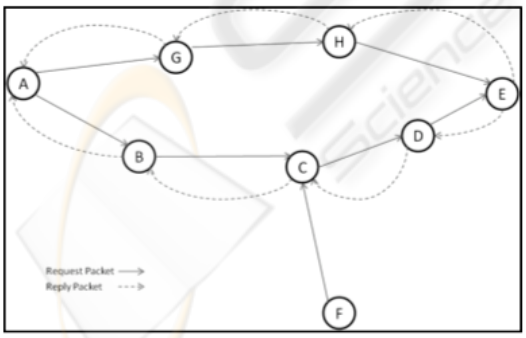


Figure 1: Illustration of how a malicious node F is detected in the system.

Population Maintenance

The number of nodes in the population of a particular node is not equal to the number of routing table entries, but equal to the number of distinct next-hops in the routing table. This is because, there might be a single node which is the next-hop for two destinations, which results in two routing table entries, but only one distinct node in the population. At some point, the members in the population may increase beyond the population size. At this juncture, the node has to eliminate certain nodes from its population, which it does by evaluating the DTF of the nodes present in the routing table and keeping the nodes with the maximum DTF out of the nodes present.

Let us illustrate this with the help of an example where the Population size is 5. In Figure 2, frame 1 shows the scenario when node *B* enters into the network and has no nodes in its population. Slowly, as it needs to communicate more, it starts finding more nodes as next-hops to include in its population. This is depicted in frames 2,3,4,5 and 6 of Figure 2. Node *B* does not check for any parameter before taking a node into its population till the population size is reached.

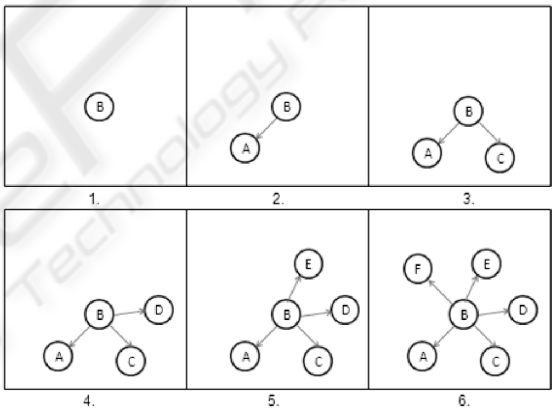
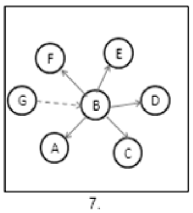
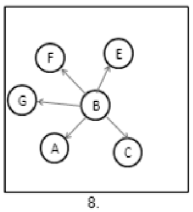


Figure 2: Shows how other nodes get included into population of node B till Population Size is reached.



Node	DTF
A	0.34
C	0.51
D	0.12
E	0.36
F	0.57
G	0.35

Table depicts the DTF calculated for all the nodes when an extra node competes to be in the population of node B



Node	DTF
A	0.34
C	0.51
E	0.36
F	0.57
G	0.35

Table depicts how node D is eliminated and how node G is included in the population of node B

Figure 3: Shows the scenario after an extra node competes to be included into node B's population.

Now in Figure 3, see frame 7 on top of figure, node *G* is also discovered by node *B,* but it already has the maximum number of members in its population, but does not know if node *G* should be included or not. To resolve this, it calculates the DTF of all the nodes in its population along with the DTF for node *G*. After calculation, node *B* finds that the node *D* has the minimum DTF and thus decides to eliminate node *D* and give membership to node *G* in its population; thus, maintaining its population size.

Updating of Next-Hops

Route maintenance is required to minimize the occurrences of route errors. In this subroutine, when a node receives a reply packet for a destination, it checks in its routing table if the current next-hop is the same as the previously stored next hop.

An updating subroutine is also used in the protocol for updating the nodes in the population of a certain node about the activity of that node. An UPDATE packet is created and transmitted to the next-hop nodes in the routing table of a node whenever there is a change in its characteristic parameters namely: packets generated, packets received and packets forwarded. This is uni-cast to only those nodes which feature as a next-hop in its routing table. It is also to be noted here, that the node sending the UPDATE packet does not send it to that entry in the routing table which it used as a next-hop for data transmission. This is beneficial since all the other nodes will have updated information regarding the node and hence a fresh value of the DTF. In this process, it is to be noted that there is no reply for the UPDATE packet in order to prevent the unnecessary usage of battery power of a node.

To prevent the network from getting flooded with update packets and also to reduce the energy consumption at the sending and receiving nodes of these packets, the update packets are sent after specific intervals rather than after every data packet. This interval is dictated by the amount of change in battery power left, number of packets transmitted, packets received and packets generated.

Route Error

This subroutine is invoked when a route existing in the routing table turns out to be broken. Once a source node needs to send data to a destination node, it looks up its routing table. If a route exists then, the packet is forwarded to the next-hop for the given destination. Similarly, the packet is forwarded till it reaches the destination. In case, the route found in the routing table is outdated or broken, this subroutine starts, where, the route-error packet traces its path back to the source, which removes the entry from its routing table and starts forward the route discovery procedure (Sanjay et al., 2010). While the route-error packet is going back to the source node from the node where the broken path was discovered, all the intermediate nodes also delete the entry for the specific destination from their routing tables.

5 SIMULATION ANALYSIS AND RESULTS

The simulation experiments conducted were evaluated using the Global Mobile Information System Simulator (GloMoSim version 2.03) (Bajaj et al., 1997). It has been designed using the parallel discrete-event simulation capability provided by PARSEC (Bagrodia et al., 1998). PARSEC is a C-based simulation language, developed by the Parallel Computing Laboratory at UCLA. It can also be used as a parallel programming language. GloMoSim currently supports protocols for a purely wireless network. In this section, we shall be comparing the DEESR protocol with the SNEP and the QDV protocols to gauge the security aspect of the DEESR protocol. A comparative study is done on the basis of their time of detection of malicious nodes and the number of malicious nodes detected out of the total malicious nodes present.

Figure 4 shows the effect of malicious nodes on the packet delivery ratio in the network. In Figure 4, the x-axis represents the percentage of malicious nodes in the network. The y-axis represents the packet delivery ratio for the three protocols. As can be seen from the Figure, the packet delivery ratio of DEESR protocol is considerably better than both the QDV protocol and the SNEP protocol. But, in general, with the increase in the percentage of malicious nodes, the

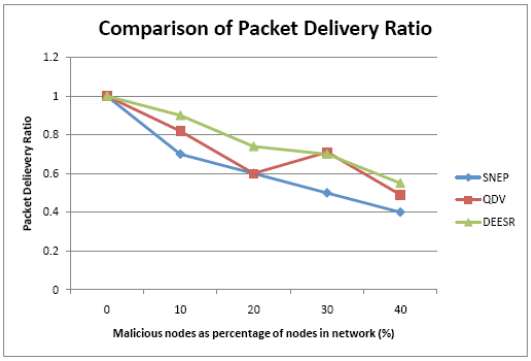


Figure 4: Comparison of Packet Drop ratio against two secure protocols.

packet delivery ratios drop significantly in the network; dropping to as low as 0.6 in the case of SNEP.

In Figure 5, we compare the detection times of the three protocols while varying the percentage of malicious nodes in the network. Thus, here also the x-axis represents the percentage of malicious nodes in the network. The y-axis represents the time taken to detect the malicious nodes in the network. It is clear from the Figure that the time taken for detection of the malicious nodes is minimum for DEESR protocol in all the instances. For both QDV and DEESR protocols, the amount of time taken to detect malicious nodes keeps decreasing when the percentage of malicious nodes keeps increasing.

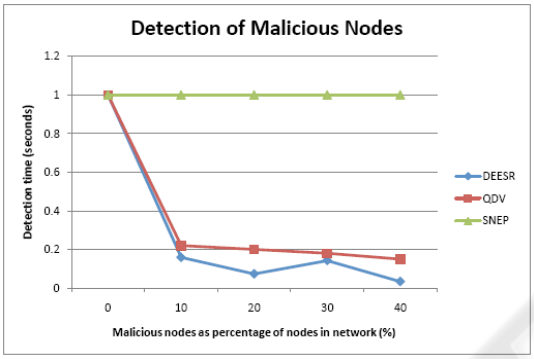


Figure 5: Comparison of time taken to detect malicious nodes in the network.

6 CONCLUSIONS

In this paper, we have discussed the two types of security threats viz. attacks by an external attacker and attacks through a compromised node. Then we discussed an approach towards secure routing which involves automatic detection and removal of malicious nodes in the system to keep the system secure while making sure that the network keeps operating despite the attacks. With extensive simulation studies, we believe that the protocol is safe for communication in different scenarios, generally delivering better results in terms of the packet delivery percentage and detection of malicious nodes over the QDV and the SNEP security based protocols. Thus, the protocol can provide reasonable amounts of services while keeping a good level of security. Since security is a dynamic field and there might be new ways of attacking being discovered every-day, the task does not end with this protocol itself. Future work involves adapting the protocol to major routing protocols and testing the protocol in a real implementation environment.

REFERENCES

Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, "Wireless sensor networks for habitat monitoring", in *Proceedings of the 1st ACM international workshop on Wireless Sensor Networks & Applications*, Atlanta, Georgia, pp. 88-97, 2002.

Ilker Demirkol, Cem Ersoy, and Fatih Alagöz, "MAC Protocols for Wireless Sensor Networks: A Survey", *IEEE Communications Magazine*, Vol. 44, Issue 4, pp. 115-121, 2006

Adrian Perrig, Robert Szewczyk, J. D. Tygar, Victor Wen and David E. Culle, "SPINS: Security Protocols for Sensor Networks", in *Proceedings of 7th Annual International Conference on Mobile Computing and Networking*, pp. 189-199, Rome, Italy, July 2001.

Sanjay K. Dhurandher, S. Misra, M. S. Obaidat and N. Gupta, "An Ant Colony Optimization Approach for Reputation and Quality-of-Service-Based Security in Wireless Sensor Networks", *Security and Communication Networks*, John Wiley & Sons, Volume 2, Issue 2, pp. 215-224, March/April 2009.

Sanjay K. Dhurandher, Mohammad S. Obaidat, Deepank Gupta, Nidhi Gupta, Anupriya Asthana, "A Dynamic Energy Efficient Routing Protocol for Wireless Ad Hoc Sensor Networks in Urban Environments", submitted to I*EEE Globecom 2010.*

S. Yi, P. Naldurg, and R. Kravets, "Security-aware routing protocol for wireless ad hoc networks," in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp 299-302, Long Beach, CA, USA, October 2001.

Chris Karlof, Naveen Sastry, David Wagner, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks", in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004*, Baltimore, MD, USA, pp. 162-175, November 3-5, 2004.

M. Dorigo and T. Stuetzle, *Ant Colony Optimization*, Prentice Hall, 2004.

L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla. "GlomoSim: A Scalable Network Simulation Environment". *Technical Report CSD Technical Report, 990027, UCLA*, 1997.

Bagrodia R., Meyer R., Takai M., Chen Y., Zeng X., Martin J., Song H.Y. "PARSEC: a parallel simulation environment for complex systems" *IEEE Computer,* Vol. 31, No. 10, pp. 77-85, 1998.