An Efficient Secure End to End Protocol in WSN

[1] Telakuntla Suma Sri, [2] Sri P. S. V. S. V. Rama Raju

[1]M.Tech (CSE) from Diet, Anakapalli, Visakhapatnam, [2]Senior Assistant Professor, Diet, Anakapalli, Visakhapatnam

Abstract: ***Now a day's wireless sensor network is most important technology for transferring data through network with secure manner. Before transferring message from source node to destination node we can find out path consisting of connected links. To identify the routing from source node to destination node so many end to end routing protocols are existing in the world. In this paper we are implementing a novel design secure end to end routing protocol for transfer data with securely. Before performing data transformation process we can implement two more fundamental concepts are user key establishment and authentication. The user authentication process enables for identify users by group key manager. The generation key we are using differ hellman key exchange algorithm. The authentication of both users we are implementing a random nonce based authentication schema. Before transferring data to destination node the source will send ids to group key manager. The server will find routing from source node to destination node, using that path data will be transferred to destination node. Before transferring message the source node will encrypt the message and send to destination node. By performing data encryption and decryption process we are using cryptography technique. So that by implementing those concepts we can improve efficiency of network and also provide more security of transferred message.***

Index Terms - Cryptography, routing, security, network security, wireless sensor network.

I. INTRODUCTION

Wireless sensor network (WSN) is widely considered as one of the most important technologies for the twenty-first century [1]. In the past decades, it has received tremendous attention from both academia and industry all over the world. A WSN typically consists of a large number of low-cost, low-power, and multifunctional wireless sensor nodes, with sensing, wireless communications and computation capabilities [2,3]. These sensor nodes communicate over short distance via a wireless medium and collaborate to accomplish a common task, for example, environment monitoring, military surveillance, and industrial process control [4]. The basic philosophy behind WSNs is that, while the capability of each individual sensor node is limited, the aggregate power of the entire network is sufficient for the required mission. In many WSN applications, the deployment of sensor nodes is performed in an ad hoc fashion without careful planning and engineering. Once deployed, the sensor nodes must be able to autonomously organize themselves into a wireless communication network. Sensor nodes are battery-powered and are expected to operate without attendance for a relatively long period of time. In most cases it is very difficult and even impossible to change or recharge batteries for the sensor nodes. WSNs are characterized with denser levels of sensor node deployment, higher unreliability of sensor nodes, and sever power, computation, and memory constraints. Thus, the unique characteristics and constraints present many new challenges for the development and application of WSNs. Our focus is on routing security in wireless sensor networks. Current proposals for routing protocols in sensor networks optimize for the limited capabilities of the nodes and the application specific nature of the networks, but do not consider security. Although these protocols have not been designed with security as a goal, we feel it is important to analyze their security properties.

When the defender has the liabilities of insecure wireless communication, limited node capabilities, and possible insider threats, and the adversaries can use powerful laptops with high energy and long range communication to attack the network, designing a secure routing protocol is non-trivial. We present crippling attacks against all the major routing protocols for sensor networks. Because these protocols have not been designed with security as a goal, it is unsurprising they are all insecure. However, this is non-trivial to fix: it is unlikely a sensor network routing protocol can be made secure by incorporating security mechanisms after design has completed.

Our assertion is that sensor network routing protocols must be designed with security in mind, and this is the only effective solution for secure routing in sensor networks. (Size 10 & Normal)This document is a template. An electronic copy can be downloaded from the conference website. For questions on paper guidelines, please contact the conference publications committee as indicated on the conference website. Information about final paper submission is available from the conference website.

II. RELATED WORK

Security issues in ad-hoc networks are similar to those in sensor networks and have been well enumerated in the literature [5], [6], but the defense mechanisms developed for ad-hoc networks are not directly applicable to sensor networks. There are several reasons for why this is so, but they all relate to the differences between sensor and ad-hoc networks enumerated in the previous section. Some ad-hoc network security mechanisms for authentication and secure routing protocols are based on public key cryptography [7], [8], [9], [10], [11], [12], [13], [14]. Public key cryptography is too expensive for sensor nodes. Security protocols for sensors networks must rely exclusively on efficient symmetric key cryptography. Secure routing protocols for ad-hoc networks based on symmetric key cryptography have been proposed [15], [16], [17], [18].

These protocols are based on source routing or distance vector protocols and are unsuitable for sensor networks. They are too expensive in terms of node state and packet overhead and are designed to find and establish routes between any pair of nodes—a mode of communication not prevalent in sensor networks. Marti et al. [19] and Buchegger and Boudec [20] consider the problem of minimizing the effect of misbehaving or selfish nodes on routing through punishment, reporting, and holding grudges. These applications of these techniques to sensor networks are promising, but these protocols are vulnerable to blackmailers. Perrig et al. present two building block security protocols optimized for use in sensor networks, SNEP and TESLA. SNEP provides confidentiality, authentication, and freshness between nodes and the sink, and TESLA provides authenticated broadcast.

Routing is another very challenging design issue for WSNs. A properly designed routing protocol should not only ensure a high message delivery ratio and low energy consumption for message delivery, but also balance the entire sensor network energy consumption, and thereby extend the sensor network lifetime Motivated by the fact that WSNs routing is often geography based, we propose a geography-based secure and efficient Cost-Aware Secure routing (CASER) protocol for WSNs without relying on flooding. CASER allows

messages to be transmitted using two routing strategies, random walking and deterministic routing, in the same framework. The distribution of these two strategies is determined by the specific security requirements. This scenario is analogous to delivering US Mail through USPS: express mails cost more than regular mails; however, mails can be delivered faster. The protocol also provides a secure message delivery option to maximize the message delivery ratio under adversarial attacks. In addition, we also give quantitative secure analysis on the proposed routing protocol based on the criteria proposed in Routing is a challenging task in WSNs due to the limited resources. Geographic routing has been widely viewed as one of the most promising approaches for WSNs. Geographic routing protocols utilize the geographic location information to route data packets hop-by-hop from the source to the destination. While geographic routing algorithms have the advantages that each node only needs to maintain its neighbouring information, and provide a higher efficiency and a better scalability for large scale WSNs, these algorithms may reach their local minimum, which can result in dead end or loops. To solve the local minimum problem, some variations of these basic routing algorithms were proposed in. In , source location privacy is provided through broadcasting that mixes valid messages with dummy messages. The main idea is that each node needs to transmit messages consistently.

III. PROPOSED SYSTEM

In this paper we proposed a novel design end to end routing protocol for finding shortest path and also provide authentication of communication entities in the network. Before performing the finding shortest route the source node and destination node will generate shared key and perform the authentication process. After completion of authentication process the source node will send destination id to server. Before performing encryption and decryption process we can find shortest route by using end to end routing protocol. After that the sender will encrypt message and convert into cipher format. The completion of encryption process the sender will send that cipher format data to destination node through the path. The destination node will retrieve that data and perform the decryption process. By performing decryption process the destination node will get original message. The implementation procedure of user's authentication is as follows.

Mutual authentication process:

After completion of building network we can perform the mutual authentication of both users in the network. By implementing process of mutual authentication is as follows.

- Now if two users U1 and U2 have become adjacent to one another, then these users are need to execute authentication process so that User U1 proves to U2 and user U2 proves to U1.
- Before performing verification process each user will choose two prime numbers p and g.
- After that each user will choose one private key (a) and calculate public key based on following formula.

  Public key= g a mod p
- After calculating public keys each user will shared those values and again will calculate shared key base on following formula.

Shared key= pub a mod p

- By calculating those shared keys are same for both users.
- After completion of shared keys each user will be verified by each other by performing following process.
- User U1 choose random nonce and send message that is received by user U2.
- User U2 also choose random nonce and send message that is received by user U1.
- After sending that random nonce user U1 will generate verification message for User U2. The generation of verification message is as follows.

Verify (U1, U2, H (n|U1|U2|shared key1)

After generating verification message that send to User U2

- The user U2 also generate verification message for User U1 and send that message to User U1.

Verify (U2, U1, H (U2|U1|n|shared key2))

After sending those verification messages each and every user will verify and both verification messages are equals those are the authenticated users. If both verification messages are not equal those are not authenticated users. After that the sender will choose the destination node id and send that id to server. By using those ids of sender and receiver the server will find out shortest route by calculating shortest distance between nodes or users or group members.

Generation of distance matrix and finding Shortest Routing:

In this module the server will generate distance matrix and finding shortest route. The implementation process of distance matrix is as follows.

- The server will get all nodes of distance points and using those points we can generate distance matrix.
- Take the each node distance points and calculate difference between each node put into matrix format. This process will repeat until completion of all nodes distance.
- The distance of each node to other node is as follow. $di = (x1-x2) + (y1 –y2)$
- Finding distance source node to other nodes by using following formula

```
int max=0;
int min=di;
if(max<min)
  {
        Max=min;
      }
```

- After finding distance of each node we can arrange the path from source node to destination node.
- So that the data send through path and reached the destination node.

After finding the path source node will transfer the data through path to destination node. Before sending data to destination node the source node will encrypt the data and transfer to destination node. The implementation procedure encryption and decryption is as follows.

Encryption Process:

In this module the sender node will enter transferred message and convert that message to unknown format. By converting plain format data into unknown format is known as encryption process. The implementation procedure of encryption process is as follows.

- The sender node will take message and key as input of encryption process.
- The sender node gets single character from message and converts into decimal value.
- Take the decimal value and key perform the xor operation until message length is completed.
- After completion of xor operation take the each decimal value and convert into eight bit binary format.
- Take the each eight bit binary data and partition into equal parts.
- Take those equal parts and reverse those binary partitions. Performing this reverse process until the message binary bits of data is completed.
- Take those binary reverse bits and generate 32 * 32 matrix format.
- Take that matrix and perform circular rotation from outer circle to inner circle.
- After completion of circular rotation read each eight bit binary format and convert into decimal value. This process continues until all matrix data is completed.

Take those decimal values as cipher format data and send to destination node through the path. The destination node will retrieve cipher format data and convert into plain format data by performing decryption process. The implementation process of decryption is as follows.

Decryption Process:

In this module the destination node will perform decryption process for converting cipher format data into plain format.

- The destination node will take cipher format data and key as input to decryption process.
- The destination node takes each decimal value from cipher data and converts into eight bit binary format data.
- Take those binary format data and generate 32 * 32 matrix format.
- Take those matrix format data and perform reverse circular rotation from outer circle to inner circle.
- After completion of circle rotation process take each eight bit binary format data and performing equal sub partition.
- Take those partitions binary data and perform the reverse process of both sub parts.
- After completion of reverse process take each eight bit binary format data and convert into decimal format until completion of cipher binary format data.
- Take decimal value and key perform the xor operation between them until completion of all decimal values.
- Take the xor data and convert into character format it will get plain format message.

By implementing those concepts we can improve the network efficiency and also provide more security of transferring message.

The previous analysis, we can establish a first upper bound on the packet propagation speed, when a classical routing strategy is employed, i.e., when packets are forwarded in "push mode" to the next relay on a hop by hop basis. For the analysis, we consider that the distribution of the signal to noise ratio is exactly known and that classical routing is optimized to achieve the fastest propagation speed under this distribution. We describe the guiding principle and the design of Opportunistic Routing with Congestion Diversity (D-ORCD). We propose a time-varying distance vector, which enables the network to route packets through a neighbor with the least estimated delivery time. D-ORCD opportunistically routes a packet using three stages of: (a) transmission, (b) acknowledgment, and (c) relaying. During the transmission stage, a node transmits a packet. During the acknowledgment stage, each node that has successfully received the transmitted packet sends an acknowledgment (ACK) to the transmitter node. D-ORCD then takes routing decisions based on a congestion-aware distance vector metric, referred to as the congestion measure. More specifically, during the relaying stage, the relaying responsibility of the packet is shifted to a node with the least congestion measure among the ones that have received the packet. The congestion measure of a node associated with a given destination provides an estimate of the best possible draining time of a packet arriving at that node until it reaches destination. Each node is responsible to update its congestion measure and transmit this information to its neighbors. Next, we detail D-ORCD design and the computations performed at each node to update the congestion measure.

Congestion Computations:

The congestion measure associated with node i for a destination d at time t is the aggregate sum of the local draining time at node i (denoted by Ldi (t)) and the draining time from its next hop to the destination (denoted by Dd i (t)), i.e. Assuming a FIFO discipline at layer-2, we proceed to decompose the local draining time. This relies on the

Observation that when a packet arrives at a node, i, its waiting time is equal to the time spent in draining the packets that have arrived earlier plus its own transmission time. If P(I;d)(t) denotes the probability that the packet transmitted by node i is successfully received by a node with lower congestion measure, then expected transmission time at node i for the packet is given by 1 Pi;d(t) .

$$V_i^d(t) = L_i^d(t) + D_i^d(t).$$

Let Qdi(t) denote the number of packets destined for destination d averaged over previous computation cycle. Qdi(t) is updated as D-ORCD computes the expected congestion measure "down the stream" for each node i to using the latest congestion. The three-way handshake procedure discussed in Section II-A to achieve receiver diversity gain in an opportunistic scheme is achieved at the cost of an increase in the control overhead. In particular, it is easy to see that this overhead cost, which is the total number of ACKs sent per data packet transmission, increases linearly with the size of the set of potential forwarders. Thus, we consider a modification of D-ORCD in the form of opportunistically routing with partial diversity (P-ORCD). This class of routing policies is parameterized by parameter M denoting the maximum number of forwarder nodes. This is equivalent to a constraint on the maximum number of nodes allowed to send acknowledgment per data packet transmission. Such a constraint will sacrifice the diversity gain, and hence the performance of any opportunistic routing policy, for lower communication overhead.

V. CONCLUSION

Our proposed system we are implementing a novel design protocol for performing authentication and key generation process. It can also implement concepts for finding shortest route from source node to destination node. In this paper we can also implement the concepts data encryption and decryption process. The authentication of users or group members can be done by group key manager and send that status to each group member. After that the group key manager will generate secret key and send that key to all group members. Each group member or user retrieve group key and send the source node, destination node to group key manager. The group key manager will retrieve source node and destination node, using those nodes ids the group key manager will calculate shortest route from source node to destination node. After finding the shortest route the group key manager send that path to both users. Both users are retrieve path and source node will encrypt the transferred message. After converting plain format data into cipher format data can be send to specified destination node. The destination node will retrieve the cipher format and perform the decryption process, it will get original plain format message. So that by proposing those concepts we can provide more security of transferring message and also improve network efficiency.

References

[1].―21 ideas for the 21st century‖, Business Week, Aug. 30 1999, pp. 78-167.
[2].S.K. Singh, M.P. Singh, and D.K. Singh, ―A survey of Energy-Efficient Hierarchical Cluster-based Routing in Wireless Sensor Networks‖, International Journal of Advanced Networking and Application (IJANA), Sept.–Oct. 2010, vol. 02, issue 02, pp. 570– 580.
[3].S.K. Singh, M.P. Singh, and D.K. Singh, "Energy-efficient Homogeneous Clustering Algorithm for Wireless Sensor Network", International Journal of Wireless & Mobile Networks (IJWMN), Aug. 2010, vol. 2, no. 3, pp. 49-61.
[4].Jun Zheng and Abbas Jamalipour, ―Wireless Sensor Networks: A Networking Perspective‖, a book published by A John & Sons, Inc, and IEEEE, 2009.
[5].L. Zhou and Z. Haas, ―Securing ad hoc networks,‖ IEEE Network Magazine, vol. 13, no. 6, November/December 1999.
[6]. F. Stajano and R. J. Anderson, ―The resurrecting duckling: Security issues for ad-hoc wireless networks,‖ in Seventh International Security Protocols Workshop, 1999, pp. 172–194.
[7]. J. Hubaux, L. Buttyan, and S. Capkun, ―The quest for security in mobile ad hoc networks,‖ in Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC 2001), 2001.
[8]. J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, ―Providing robust and ubiquitous security support for mobile ad-hoc networks,‖ in ICNP, 2001, pp. 251– 260.
[9]. M. G. Zapata, ―Secure ad-hoc on-demand distance vector (SAODV) routing,‖ IETF MANET Mailing
List, Message-ID: 3BC17B40.BBF52E09@nokia.com, Available at ftp://manet.itd.nrl.navy.mil/pub/manet/2001-10.mail, October 8, 2001.
[10] H. Luo, P. Zefros, J. Kong, S. Lu, and L. Zhang, ―Self-securing ad hoc wireless networks,‖ in Seventh
IEEE Symposium on Computers and Communications (ISCC ’02), 2002.
[11]. J. Binkley and W. Trost, ―Authenticated ad hoc routing at the link layer for mobile systems,‖ Wireless
Networks, vol. 7, no. 2, pp. 139–145, 2001.
[12]. B. Dahill, B. N. Levine, E. Royer, and C. Shields, ―A secure routing protocol for ad-hoc networks,‖
Electrical Engineering and Computer Science, University of Michigan, Tech. Rep. UM-CS-2001-037, August 2001.
[13]. J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu, ―Adaptive security for multi-layer ad-hoc networks,‖ Special Issue of Wireless Communications and Mobile Computing, Wiley Interscience Press, 2002.
[14]. Y.-C. Hu, D. B. Johnson, and A. Perrig, ―SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks,‖ in Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002), June 2002, pp. 3–13.
[15]. Y.-C. Hu, A. Perrig, and D. B. Johnson, ―Ariadne: A secure on-demand routing protocol for ad hoc networks,‖ Department of Computer Science, Rice University, Tech. Rep. TR01-383, December 2001.