Trust Management Problem in Distributed Wireless Sensor Networks

Riaz Ahmed Shaikh[1], Hassan Jameel[1], Sungyoung Lee[1], Saeed Rajput[2] and Young Jae Song[1]
[1]*Department of Computer Engineering, Kyung Hee University, Republic of Korea*
[2]*Department of Computer Science and Engineering, Florida Atlantic University, USA*
{riaz, hassan, sylee}@oslab.khu.ac.kr, srajput@fau.edu, yjsong@khu.ac.kr

Abstract

Sensor network security solutions that have been proposed so far are mostly built on the assumption of a trusted environment, which is not very realistic so we need trust management before deploying any other security solution. Traditional trust management schemes that have been developed for wired and wireless ad-hoc networks are not suitable for wireless sensor networks because of higher consumption of resources such as memory and power. In this paper, we propose a novel lightweight group based trust management scheme (GTMS) for distributed wireless sensor networks in which the whole group will get a single trust value. Instead of using completely centralized or distributed trust management schemes, GTMS uses hybrid trust management approach that helps in keeping minimum resource utilization at the sensor nodes.

1. Introduction

Wireless sensor networks are susceptible to various types of security threats such as eavesdropping, message replay, and fabrication of messages. These threats can be avoided by introducing various safety mechanisms such as authentication, confidentiality, and message integrity. These safety mechanisms are dependent upon cryptographic schemes that need robust and secure key exchange mechanism. If the key exchange mechanism is securely carried out successfully, we say that the two nodes have established "Trust" in each other. If one or multiple communicating nodes are compromised before the successful key exchange, any subsequent safety mechanisms are rendered ineffective. Thus there is a clear need to establish trust between communicating nodes. So, to establish secure communications, we need to ensure that all communicating nodes are trusted. That's why trust establishment is a prerequisite of any security implementation and both are tightly interdependent.

This work is financially supported by the Ministry of Education and Human Resources Development (MOE), the Ministry of Commerce, Industry and Energy (MOCIE) and the Ministry of Labor (MOLAB) through the fostering project of the Lab of Excellency. The corresponding author of this paper is Prof. Sungyoung Lee.

Current research on sensor network security is mostly built on the assumption of a trusted environment [1]. Security solutions such as SPINS [2], TinySec [3], LiSP [4], and LSec [5] etc that have been developed so far are based on this same assumption. Traditional trust management schemes [6, 7, 8, 9, 10] that have been developed for wired and wireless ad-hoc networks are not suitable for wireless sensor networks because of higher consumption of resources such as memory and power as we will discuss here. Therefore we need a lightweight trust management scheme for large scale distributed wireless sensor networks.

Trust management schemes can be either centralized or distributed, but we believe that neither completely centralized nor completely distributed trust management schemes are suitable for wireless sensor networks. Centralized trust schemes are not appropriate because they are energy expensive due to extra routing overhead. In large sensor networks, the total routing cost for the exchange of trust values of a sensor node with the base station is quite energy expensive when the base station is far away from the node. Totally distributed approaches are also not suitable because each node has limited memory and computation power. In a distributed approach, each node needs to maintain the up-to-date record about the trust values of entire network in the form of a database. The size of the database is directly proportional to the size of the network. It is not possible for a single sensor node to store and compute the trust values of the entire network. Therefore some hybrid scheme is needed.

We also believe that sensor nodes mostly fulfill their responsibilities in a cooperative manner [11] rather than individually. Therefore instead of calculating individual trust, it is more appropriate to calculate the trust for the entire group.

Research on trust management scheme for wireless sensor networks is in the infancy state. Hence, in this work, we propose a novel lightweight group based trust management scheme (GTMS) for distributed wireless sensor networks that is based on a hybrid trust management scheme. Rest of the paper is organized as follows: Section 2 describes the Group based trust management scheme. Section 3 consists of conclusion and future directions.

2. Group based Trust Management Scheme (GTMS)

Our trust model is based upon hybrid trust management scheme. Within a group we used distributed trust management approach in which all sensor nodes need to calculate individual trust values for all group members. Cluster head will aggregate these trust values and forward it to the base station (BS). Then, the base station will calculate the cumulative trust value of the whole group. Depending upon that trust value, BS will assign one out of three possible states, namely: trusted, un-trusted and un-certain to the whole group as shown in figure 1. In this way, the state of all the groups will be calculated and stored at the base station. After that, the base station will periodically multicast the current state of each group to all cluster heads. Here BS is the central authority.

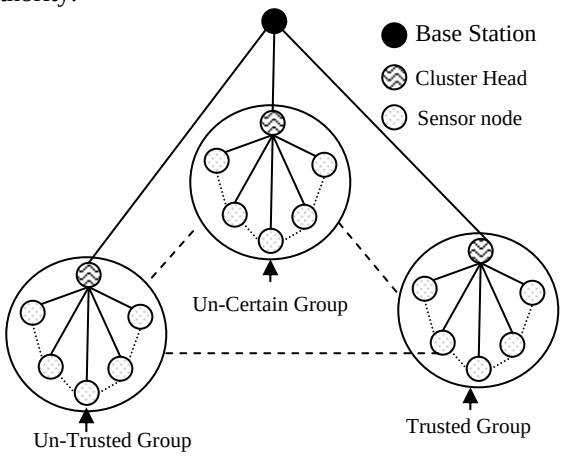

Fig 1: Conceptual Group based Trust Model Scenario

Our group based trust model works in three phases

1. Trust Calculation at Node
2. Trust Calculation at Cluster Head
3. Trust Calculation at Base Station

The trust value Tv_{xy} of node y calculated by node x is a value between 0 and 1.

2.1. Trust Calculation at Node

At the Node level, trust value is calculated by using time based past interaction as well as peer recommendations.

2.1.1. Time based Past Interactions Evaluation: Time based past interaction value of node y at node x, $PI_{x,y}$ is defined as:

$$PI_{x,y} = 1 - \frac{1}{\max[\{w_s SI_{x,y} - w_u UI_{x,y}\}, 0] + 1} \quad \cdots (1)$$

Where $PI_{x,y}$ is the past interaction value of y calculated by node x based upon past interactions, $SI_{x,y}$ is the successful interactions of nodes x with y, $UI_{x,y}$ is the unsuccessful interactions of node x with y. w_s and w_u are positive numbers (depending upon time) and represent the corresponding weights of SI_{xy} and $UI_{xy.}$ The weight w_s is defined as:

$$w_s = \left\{ \begin{array}{ll} h & t_s = 1 \\ m & t_s = 2 \\ l & t_s > 2 \end{array} \right\} \cdots (2)$$

w_u is defined similarly. This mapping function helps to assign higher, medium or low weights based on the last interaction time. The time t_s and t_u are defined as:

$$t_s = \left\lceil \frac{T_{current} - ST_{x,y}}{\Delta T} \right\rceil, t_u = \left\lceil \frac{T_{current} - UT_{x,y}}{\Delta T} \right\rceil \cdots (3)$$

Where $T_{current}$ is the current time, $ST_{x,y}$ ($UT_{x,y}$) is the time of last successful (unsuccessful) interaction and ΔT is the threshold time.

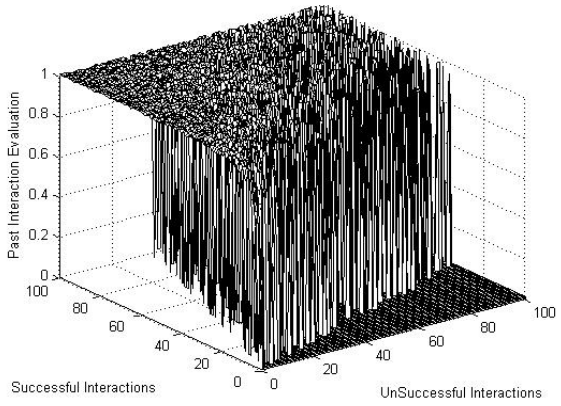

Fig 2: Time based Past Interactions Evaluation

Figure 2, shows the graph of the past interaction evaluation against successful and unsuccessful interactions. 'h', 'm' and 'l' were given the values 3, 2 and 1 and w_s and w_u were randomly assigned one of these values in every computation of $PI_{x,y}$. The graph shows fluctuations when SIx,y and UIx,y have roughly the same values but when SIx,y is considerably larger than UIx,y , $PI_{x,y}$ approximates to 1.

2.1.2. Peer Recommendations Evaluation: Let's suppose there are n nodes in the group and each node has its own unique id. Each node has a trust value for any other node it interacted with before. When any node gets peer recommendations then it calculates the trust value by using the following formula.

$$PR_{x,y} = \frac{\sum_{i=1}^{n-1} [Tv_{x,i} * Tv_{i,y}]}{n-1} \quad \cdots (4)$$

Where $PR_{x,y}$ is the peer recommended trust value of node y calculated by node x, $Tv_{x,i}$ is the trust value of node 'i' calculated by node x and $Tv_{i,y}$ is the trust value of node y sent by node i.

2.1.3. Formation of Trust Value: Let's suppose node x wants to calculate the trust value ($Tv_{x,y}$) of node y, then it can be calculated by the following equation.

$$Tv_{x,y} = \frac{\left(PI_{x,y} + PR_{x,y}\right)}{2} \quad \cdots (5)$$

2.1.4. Memory Requirement at Sensor Node: Each node maintains a small trust database as shown in table 1. The size of each record is 22 bytes. Therefore memory requirement for GTMS at each sensor node is *(n-1)*22* bytes, where n is number of nodes in the cluster.

Table 1: Trust Database at Sensor Node

Node ID	Past Interactions				Peer Recommen-dations	Trust value
	$SI_{x,y}$	$ST_{x,y}$	$UI_{x,y}$	$UT_{x,y}$		
2 bytes	2 bytes	4 bytes	2 bytes	4 bytes	4 bytes	4 bytes

Size of trust database at each node, is dependent upon size of the cluster. For instance, if we assume that there are 10 nodes in the cluster then the size of trust database requires 198 bytes of memory space.

2.2. Trust Calculation at Cluster Head

Here we assume that Cluster Head is the sensor node that has higher power and memory as compared to other sensor nodes. For memory efficiency it is recommended that the size of cluster should be small and each node should directly communicate with its cluster head via single hop [12]. For the calculation of trust, cluster head broadcasts a request in a group. In response, all group member nodes forward their trust values of other member nodes to the cluster head. The trust vector of cluster head node $\overrightarrow{Tv_{ch}}$ is defined as

$$\overrightarrow{Tv_{ch}} = (Tv_{ch,1}, Tv_{ch,2}, \ldots, Tv_{ch,n}) \quad \cdots (6)$$

Where $Tv_{ch,i}$ represents the trust of node i. It is calculated as

$$Tv_{ch,1} = \frac{\sum_{i=1}^{(n-1)} Tv_{i,1}}{(n-1)}, Tv_{ch,2} = \frac{\sum_{i=1}^{(n-1)} Tv_{i,2}}{(n-1)}, .., Tv_{ch,n} = \frac{\sum_{i=1}^{(n-1)} Tv_{i,n}}{(n-1)} \cdots (7)$$

This Trust vector is forwarded to BS.

During group-to-group communications cluster head maintains the record of past interactions of another group in the same manner as individual nodes keep record of other nodes. For group based trust calculation we have adopted centralized trust based management scheme. Trust value of any group is calculated on the basis of past interaction and information sent by base station. Here we are not considering peer recommendations in order to save memory and power computation of cluster head node.

2.2.1. Formation of Trust Value: Let us suppose cluster head 'I' wants to calculate the trust value ($Tv_{i,j}$) of another cluster 'j', then it can be calculated by the following equation.

$$Tv_{i,j} = \frac{\left(PI_{i,j} + BR_{i,j}\right)}{2} \quad \cdots (8)$$

where $BR_{i,j}$ is the recommendation sent by base station to cluster head 'i' for 'j'. This trust value is later forwarded to the base station on request.

2.2.2. Memory Requirement at Cluster Head: Cluster head maintains two databases; one is similar to individual sensor node's trust database and the second one maintains the trust values of other groups as shown in table 2. The size of each record is 22 bytes. So, the total size of table 2 is *(m-1)*22*. Here 'm' is the total number of groups in the network. In order to store all the trust values in both databases, cluster head needs *(n+m-2)*22* bytes of memory space. For instance, if we assume that there are 10 nodes in the cluster and 20 groups in the network then the cluster head needs 616 bytes of memory to store these values.

Table 2: Group Trust Database at Cluster Head

Group ID	Past Interactions with Groups				Recommen--dations from BS	Trust value
	$SI_{x,y}$	$ST_{x,y}$	$UI_{x,y}$	$UT_{x,y}$		
2 bytes	2 bytes	4 bytes	2 bytes	4 bytes	4 bytes	4 bytes

2.3. Trust Calculation at Base Station

Suppose there are m groups in the network. BS periodically multicasts request and response packets to the cluster heads. On a request, cluster head forwards their trust vectors and recommendations of other groups based upon past interactions to BS. On the basis of these

responses, BS maintains the trust matrix (TM) as shown below.

$$TM = \begin{bmatrix} Tv_{G1,G1} & Tv_{G1,G2} & \cdots & Tv_{G1,Gm} \\ Tv_{G2,G1} & Tv_{G2,G2} & \cdots & Tv_{G2,Gm} \\ \vdots & \vdots & \cdots & \vdots \\ Tv_{Gm,G1} & Tv_{Gm,G2} & \cdots & Tv_{Gm,Gm} \end{bmatrix} \cdots (9)$$

Based on this matrix, it calculates the trust value of each group as shown below:

$$Tv_{BS,G1} = \frac{\sum_{i=1}^{m} Tv_{Gi,G1}}{m}, Tv_{BS,G2} = \frac{\sum_{i=1}^{m} Tv_{Gi,G2}}{m}, .., Tv_{BS,Gm} = \frac{\sum_{i=1}^{m} Tv_{Gi,Gm}}{m} \cdots (10)$$

After calculating each group's trust value, BS will map these values with the mapping function Mp for identification of state, defined below:

$$Mp(Tv_{BS,Gi}) = \begin{cases} trusted & 0.6 \le Tv_{BS,Gi} \le 1 \\ uncertain & 0.4 \le Tv_{BS,Gi} < 0.6 \\ untrusted & 0 \le Tv_{BS,Gi} < 0.4 \end{cases} \cdots (11)$$

Base station is the command center of sensor network and doesn't have constraints of limited memory and power. Therefore, we can safely ignore the issue of memory storage requirements for GTMS at the base station.

3. Conclusion and Future Directions

In this paper, we have proposed a novel group based trust management scheme (GTMS) for distributed wireless sensor networks, which is hybrid in nature. GTMS is very simple and flexible and doesn't require large storage of data and complex computations at sensor nodes. In future, we will incorporate intrusion tolerant intelligence in GTMS, so that nodes are able to detect false trust values sent by any malicious node. We will also perform a detailed simulation analysis to get the actual level of energy consumption. GTMS trust model is based on the assumption that every node has a unique id, but the following still remains an open problem: "how to assign identities and trust values to sensor nodes in a large scale anonymous environment where nodes have no unique id?"

4. Reference

[1] Elaine Shi and Adrian Perrig, "Designing Secure Sensor Networks", *IEEE Wireless Communications*, vol. 11(6), Dec 2004, pp. 38-43

[2] Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security protocols for sensor networks", *in Proc. of 7th annual international conference on Mobile computing and networking*, Rome, Italy, Aug 2001, pp 188-189

[3] Chris Karlof, Naveen Sastry, and David Wagner, "TinySec: a link layer security architecture for wireless sensor networks", *in Proc. of the 2nd international conference on Embedded networked sensor systems*, Baltimore, MD, USA, Nov 2004, pp 162-175

[4] Taejoon Park, and Kang G. Shin, "LiSP: A Lightweight Security Protocol for Wireless Sensor Networks", *ACM Transactions on Embedded Computing Systems*, vol. 3(3), Aug 2004, pp. 634–660

[5] Riaz A. Shaikh, Sungyoung Lee, M. A. U. Khan and Young Jae Song, "LSec: Lightweight Security Protocol for Distributed Wireless Sensor Network", *will appear in Proc. of 11th IFIP International Conference on Personal Wireless Communication (PWC'06), Spain, Sep* 2006

[6] Abdul-Rahman, "The PGP Trust Model", *EDI-Forum: the Journal of Electronic Commerce*, 1997, April, 1997

[7] Abdul-Rahman and Stephen Hailes, "A Distributed Trust Model", *in Proc. of ACM New Security paradigms workshop*, 1997, pp. 48-60

[8] N. Li, J. Mitchell, and W. Winsborough, "Design of a role based trust management framework", *in Proc. of the IEEE Symposium on Security and Privacy*, Oakland, 2002, pp. 114-130

[9] Zhaoyu Liu, Anthony W. Joy, and Robert A. Thompson, "A Dynamic Trust Model for Mobile Ad Hoc Networks", *in Proc. of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS'04)*, May 2004, pp 80-85

[10] Asad Amir Pirzada and Chris McDonald, "Establishing Trust in Pure Ad-hoc Networks", *in Proc. of 27th Australasian Computer Science Conferenc*e, Dunedin, New Zealand, 2004, pp. 47-54

[11] S. Tilak, N. B. Abu-Ghazaleh and W. Heinzelman, "Taxonomy of Sensor Network Communication Models", *Mobile Computing and Communication Review* 6(2): 1-8, Apr 2002

[12] Ossama Younis and Sonia Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad-hoc Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 3(4), pp. 366-379, Oct-Dec 2004