Improving Key Management Scheme Supporting Node Mobility in Heterogeneous Sensor Networks using Collision-Keys

Vijay Kumar
Department of Computer Engineering
National Institute of Technology
Kurukshetra (India)

Priyanka Ahlawat, IEEE Member
Assistant Professor
Department of Computer Engineering
National Institute of Technology
Kurukshetra (India)

ABSTRACT

Wireless Sensor Networks (WSNs) consist of a large number of sensor nodes that are batteries powered, equipped with limited memory and computational capabilities. These constrained devices face many security threats and thus there is a need of some cryptographic mechanism for secure communication. Key distribution is of critical importance to provide security in WSNs. Till now a number of key distribution schemes are proposed in the literature but there are very few schemes considering mobility of sensor nodes. In this paper we have proposed a modification to the key management scheme supporting node mobility in heterogeneous sensor network. Our modification uses Hash collision keys to improve the network resilience and connectivity between the nodes. We have evaluated our scheme analytically and obtained results show that our proposed solution assures better network connectivity and resilience while increasing an insignificant computational overhead.

Keywords

Wireless Sensor Network (WSN), Heterogeneous sensor network, Resiliency, Auxiliary Node (AN), Mobile node, Key management.

1. INTRODUCTION

A Wireless Sensor Network (WSN) is a wireless network which is composed of a set of autonomous sensor nodes having sensing, computation and communication capabilities. The main purpose of such networks is to collect the information issued from a controlled environment and then send it to the Base Station (BS) for further processing. Transmission between the sensors is done by short range radio communications. Such networks are used in many applications including tracking of object in an enemy's area for military purposes, industry automation, tracking patients, wildlife monitoring, pollution tracking, monitoring fire and nuclear power plants, engineering and medical exploration, environment monitoring etc. [1]. There are two main characteristics of WSNs that introduce various security threats. First, usually a large number of sensor nodes are deployed in the network and thus for the economic reasons, sensor nodes are often highly resource constrained with tiny memory, limited computational capability and lacking tamper-resistant components. Second, sensor nodes are generally deployed in an uncontrolled and hostile environment, thus vulnerable to capture by an adversary. These two characteristics may incur various attacks such as, eavesdropping, node replication, replay attack etc. [2]. To provide security from many of such threats, data collected by sensor nodes need to be encrypted before transmitting to neighbour nodes and BS. This encryption of sensor readings should be done using symmetric cryptographic secret keys. Due to resource limitations, as well as vulnerability to physical capture by an adversary, traditional public-key ciphers such as RSA, ElGamal Cryptosystem, Elliptic Curve Cryptography (ECC), Diffie-Hellman key exchange are too complicated, energy consuming and thus infeasible for large scale sensor networks. Hence symmetric key ciphers are the doable option for encryption of sensed data. However setting up a symmetric key among communicating sensor nodes is a big challenge in sensor networks [3].

A number of symmetric key establishment schemes scheme are proposed in the literature but most of these scheme consider only static nature of sensor nodes. However there are number of applications such as military operations, wildlife monitoring, logistics, assisted living, transportation etc. which demands mobile sensor nodes [4]. This mobility can be provided in an efficient way by taking Heterogeneous Sensor Networks (HSNs). HSNs are those sensor networks which incorporate a mixture of sensor nodes with widely varying capabilities. Studies shows that heterogeneity can improve the network performance and network lifetime with an insignificant increase in cost [6].

In this paper we have proposed an improvement to the scheme proposed by Sarmad et al. [4]. In our scheme AN can create the encryption key (for sending the communication key) using the authentication keys directly shared with MN or using the collision authentication keys 'C', if the nodes (AN and MN) carry different but related keys K_a and $K_{a'}$ satisfying the condition

$$C = H(K_a) = H(K_{a'}) \quad (1)$$

where 'H' is a suitable cryptographic hash function. Now, probability of satisfying the condition (1) by two randomly chosen keys is very less, so we generate the authentication key pool in accordance with the birthday paradox scheme proposed in paper [5]. This modification improves the resilience and connectivity of the network without significant increase in computational overhead.

This paper is organized as follows: Section-2 discuss various key management schemes. Section-3 gives the description of underlying network model. Section-4 presents our proposed scheme followed by description of possible attacks and their solution in section-5. We analyze our scheme in Section-6 and Section-7 concludes this paper.

2. RELATED WORK

A number of key management schemes are proposed in the literature for WSN. In this section we provide a overview of some of the schemes for homogeneous and heterogeneous network. Eschenauer and Gligor first proposed a Random Key Pre-distribution scheme (EG Scheme) [7]. In this scheme each sensor node is assign equal number of keys, from a large key pool. Nodes having a share key can establish a secure link. If nodes do not have a share key, they establish a path key using the neighbours with whom they share a key. Connectivity of the network can be increased by decreasing key pool size or by increasing the keys stored in a MN, but this also decrease the resilience. An improvement is given by Chain in [8]. In this scheme nodes can establish a link if they share at least **q** keys. Pair-wise session key is created by applying hash function on concatenation of all the shared keys. This scheme provides better resilience but connectivity decreases as value of **q** increases. In [9], common keys are used to establish multiple logical paths over which costly threshold key sharing scheme is used to agree on a secret. Du and Lin [10] proposed a differentiated coverage algorithm for heterogeneous sensor networks. It can provide different degree of coverage according to the requirement of application. Yarvis et al. [6] analyzed the impact of heterogeneity on reliability and lifetime of the network. It has been shown that proper placement of few heterogeneous resources can significantly improve the lifetime of the network. Traynor [11] give the asymmetric pre-distribution scheme for HSN, which significantly reduce the memory overhead and provide better security. Tools for simulation, visualization and measurements for HSN are developed [12], which are critical to address the inevitable problem that crop up in deployment. A tree based key management scheme for heterogeneous sensor networks is proposed [13], which handles various events like node addition, node compromise and key refresh at regular intervals.

3. NETWORK MODEL

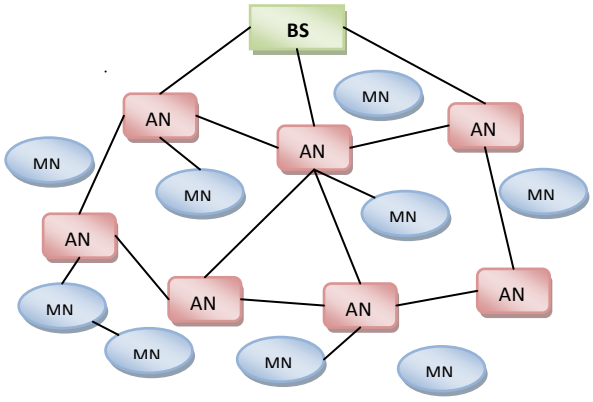

Fig 1: Network Model

Network Model that we consider is same as proposed in the paper [11]. In this model instead of considering all the sensor nodes of same type, we consider two types of nodes named Auxiliary Nodes (ANs) and Mobile Nodes (MNs). These nodes differ in terms of their capability and work they have to do. ANs are powerful nodes having more memory, processing capability and additional radios. These nodes act as a router, as they route the information collected by the MNs to the Base Station (BS). ANs are small in number and their position is assumed to be fixed, whereas MNs can change their location and are responsible for collecting the data. We also assume that keys of the ANs cannot be compromised even if they get captured. In this model MNs acts as end devices and must be authenticated before they can route the information to the BS through auxiliary nodes. Authentication is done by sharing of authentication keys between ANs and MNs. Once a node get authenticated a communication key is given by the AN to send the collected data.

Considering such a heterogeneous network model has a number of benefits, like the main problem of high energy consumption of the nodes near the sink has been removed, as in this model more powerful auxiliary nodes are responsible for routing the data. Sensor nodes can change their position and still can send the data easily until they themselves or their neighbours are in communication range of any AN.

4. Proposed Scheme

Our proposed scheme is an improved version of the scheme proposed by Sarmad et al. in [4]. It consist of three phases, key pre-distribution, MN authentication and communication key generation. Scheme is described as follows.

Notations that are used in this paper are given in the Table 1

Table 1

Symbol	Description
|P|	Size of authentication pool
S, K	Keys assign to AN and MN respectively(S>>K)
n1, n2	Disjoint key rings to generate communication key
K_{prvt}, K_{public}	Network private and public key respectively
q	Number of authentication keys used to generate the encryption key
x	Number of compromised nodes

4.1 Key Distribution before deployment

In our scheme, BS has a key pool of size |P| used for authentication between ANs and MNs. This authentication key pool is constructed as proposed by Vashek et al. in [5]. Where instead of randomly selecting |P| keys, we take |P|/2 colliding key pairs (using birthday paradox) to form the key pool |P|. BS selects key rings of size S and K (where S>>K) from this authentication key pool and assign them to ANs and MNs respectively. These key rings are selected in such a way that no colliding key pair is present in same key ring. BS also has another key pool from which it selects the key rings of size n_1 and n_2 and then assigns them to AN. These key rings are used for communication key generation. ANs are high power nodes and their position in network is assumed to be fixed. They communicate with other ANs and BS through public key cryptography. Information stored in ANs can be summarised as:

- Authentication key pool of size |S|
- Key rings of size n_1 and n_2 (for communication key generation)

- Public key of other ANs and BS
- Network private key (K_{prvt})
- Key ids of the keys stored in MNs
- Information of colliding key pairs
- Hash function (H)

4.2 MN Authentication

After the network deployment phase, MNs are authenticated through the ANs. To perform the authentication, AN must share at least q-authentication keys with MN. These keys can be the direct shared keys or the collision keys. Probability of sharing exactly *i*-direct keys and *j*-collision keys between a MN and AN can be given as in [5].

$$P_{share\ exactly}\ (i,j) = \frac{\left\{ {|P| \atop S} \right\} \binom{S}{i} \binom{S-i}{j} \left\{ {|P|-2S \atop K-i-j} \right\}}{\left\{ {|P| \atop S} \right\} \left\{ {|P| \atop K} \right\}} \qquad (2)$$

Here collision keys are counted only if their pre-images are not counted. Now, probability of sharing at least q-authentication keys (or generating the encryption key) can be calculated as follows:

$$P_{encrypt\ \ key\ generation} = \sum_{i=0}^{K} \sum_{j=0}^{K} P_{share\ exactly}\ (i,j) \qquad (3)$$

Where $i + j \geq q, i + j \leq K$

Whole authentication process can be described as follows:

- MN sends authentication request to AN by sending its node ID, encrypted by network public key (K_{public}).
- AN decrypt the request using corresponding private key (K_{prvt}).
- After that AN sends an authentication nonce encrypted by encryption key (which is a q-composite key of authentication keys, shared with MN). AN also sends the ids of the keys used in creating encryption key. If collision authentication keys are used than AN sends the id of corresponding colliding key and set a flag indicating that key is used as a collision key (ie. after applying hash function).
- If AN doesn't have enough authentication keys (ie. keys shared are < q) for encryption, than it consult BS which has complete key information of all MNs.
- After receiving the message, MN also create the encryption key using key id information send by AN, and decrypt the authentication message.

When MN changes its location and come into the range of other AN, it sends the id of previous AN to this new node. New AN get the information of incoming node from previous AN to reduce the broadcast overhead. Here ANs communicate with each other using public key cryptography. After that same authentication process is repeated to authenticate this new MN.

4.3 Communication key generation

When MN get authenticated, a communication key is given to that node for further communication with AN. This communication key is generated using the formula:

$$communication\ key = \binom{any\ key\ from}{key\ ring\ n_1}^{q}\ mod \binom{any\ key\ from}{key\ ring\ n_2} \qquad (4)$$

Here q is the number of authentication keys used in generating the encryption key. Using the formula of (4), large number of communication keys can be generated with moderate number of keys in key rings n1 and n2.

5. ATTACKS AND THEIR SOLUTION

Sensor nodes are often deployed in unattended and inconsiderate environments to perform various monitoring tasks. As a result WSNs are susceptible to many attacks. Here we consider two famous attacks named node replication and replay attack.

Node replication attack: In this attack an adversary prepares his own low cost sensor nodes and deceives the Network into accepting them as authentic ones. To do so the adversary only needs to physically capture one node, extract its secret credentials, reproduce the node in large quantity and then deploy the replicas under her control into the network possibly at strategic positions [14]. To avoid this attack each AN also send the node IDs of all the MNs with whom it is communicating, to the BS. This information helps the BS to keep the record of every MN in the network and thus helps to avoid node replication attack.

Replay attack: An adversary that eavesdrop a legitimate message sent between two authorized nodes and replays it at some later time engages in a replay attack. This attack can be avoided by using the time stamps. MNs include the time stamp value in the message encrypted by the communication key. AN decrypt the message and check the time stamp value. If valid time stamp than accept message otherwise rejected.

6. ANALYSIS

In this section we analyze our propose scheme in terms of network resilience, computation overhead and network connectivity. Comparison of results shows that proposed scheme provides better network connectivity and resilience with an insignificant increase in computational overhead.

6.1 Resilience

Resilience is defined as probability of compromising the communication between uncompromised nodes using the keys of compromised nodes. In our scheme we assume that only the keys of MN can be compromised. Now, key ring of each MN contains K-authentication keys and one communication key. Probability of using the same communication key by another MN is very less as a large number of communication keys is possible. However same authentication keys can be used for generating the encryption key, which is used to send the communication key. Thus compromise of authentication keys can also affects the communication between uncompromised nodes. Probability of fraction of communication compromised when *x*-nodes get compromised is calculated as (derivation is given in [5]):

$$P[Compromise] = \sum_{i=0}^{K}\sum_{j=0}^{K}\left(1-\left(1-\frac{K}{|P|}\right)^{x}\right)^{i}\left(1-\left(1-\frac{2K}{|P|}\right)^{x}\right)^{j} \times \frac{P_{share\ exactly\ (i,j)}}{P_{encrypt\ \ key\ \ generation}}$$

Figure1 shows the comparison of our proposed scheme with Sarmad-schme. While keeping the same connectivity level, it's clear from the graph that, our scheme provides better node capture resilience.

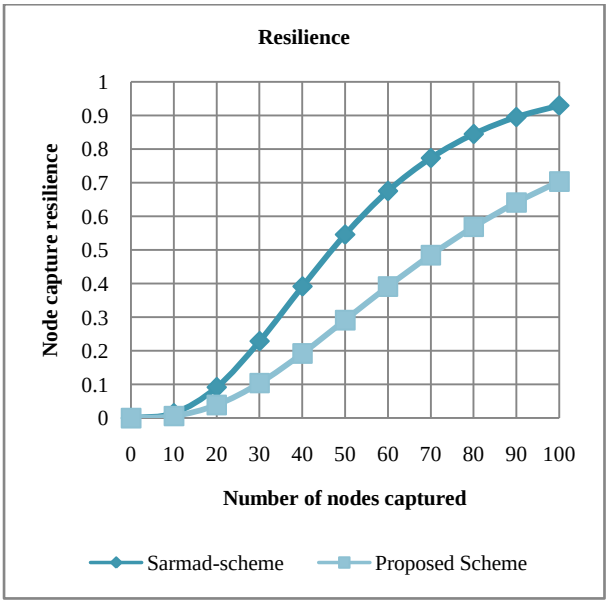

Fig 2: Node capture resilience after *x* random nodes are captured

6.2 Computation Overhead

Our Proposed scheme requires a little additional computation. This computation is equal to the number of times nodes need to apply the hash function, while creating the encryption key. For example, assume that encryption key is created using 6 keys (ie. q=6) out of which 4 are direct shared keys and 2 are collision keys, than additional computation is equal to the four hash function computation (two for encryption key and two for corresponding decryption key).

6.3 Connectivity

Connectivity of the network depends upon the key sharing probability between the nodes. Network model that we have considered provides 100% connectivity between AN and MN, as AN can consult the BS, which has complete key information of all the MNs. However if we avoid consulting the BS, than figure-2 shows that our scheme provides a better network connectivity than Sarmad-scheme, while keeping same number of keys in MNs. Our scheme also increases sharing probability within the MNs, which helps in link establishment when MN is not in direct range of any AN.

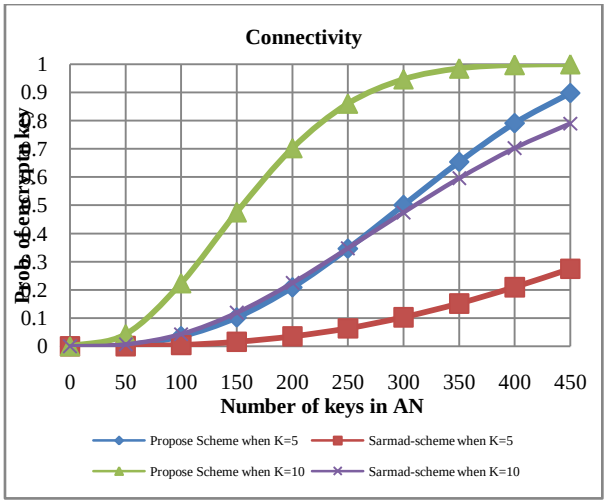

Fig 3: Probability of encryption key generation

7. CONCLUSION

In this paper we have proposed and analyzed, hash collision key improvement described in [5], to Sarmad-scheme for heterogeneous sensor networks. This modification requires a little increase in computational overhead. Results show that our proposed scheme is better in terms of connectivity and resilience. We have proposed hash collision key improvement to Sarmad-scheme, this improvement can be combined with Key-chain improvement, to further enhance the resilience and connectivity. We leave this combination for future work.

8. REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci "Wireless Sensor Networks: a survey," Computer networks 38(4):393-422, 2002

[2] H. Zhao, J. Hu, J. Qin, V. Varadharajan and H.Wan, "Hashed Random Key Pre-Distribution Scheme for Large Heterogeneous Sensor Networks," IEEE 11th Inter. Conf. on Trust, Security and Privacy in Computing and Communications, 2012.

[3] A.K. Das, "Improving identity-based Random key Establishment Scheme for large scale hierarchical Wireless Sensor Networks," International Journal of Network Security 13 (3):181-201, 2011.

[4] S.U. Khan, L. Lavagno and C. Pastrone, "A key management scheme supporting node mobility in Heterogeneous Sensor Networks," IEEE 6th inter. Conf. on Emerging Technologies (ICET), 2010.

[5] Jiri Kur, Vashek Matyas and P. Svenda, "Two Improvements of Random Key Predistribution for Wireless Sensor Networks," in Proceedings of the 8th International Conference on Security and Privacy in Communication Networks, SecureComm 2012.

[6] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu and S. Singh, "Exploiting the heterogeneity in sensor networks", INFOCOM'05,2005.

[7] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. ACM Conf. Computer and Communication Security*, pp: 41–47, 2002.

[8] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. IEEE Symp. Security and Privacy (S&P'03)*, 2003.

[9] S. Zhu, S. Xu, S. Setia, and S. Jajodia, "Establishing pairwise keys for secure communication in ad hoc networks: A probabilistic approach," in *Proc. IEEE ICNP*, 2003.

[10] X. Du, F. Lin, "Maintaining differentiated coverage in Heterogeneous sensor networks", EURASIP Journal on Wireless Communication and Networking, (4):565-572, 2005.

[11] P. Traynor, R. Kumar, H. Bin Saad, G. Cao, T. La Porta, "Efficient hybrid security mechanism for heterogeneous sensor networks," IEEE Trans. On Mobile Computing 6(6):663-677,2007.

[12] L. Girod et al, "A system for simulation, emulation and deployment of heterogeneous sensor networks", in Proceedings of ACM SenSys, 2004.

[13] A.S. Poornima, B.B. Amberker, "Tree-based Key Management Scheme for Heterogeneous Sensor Networks". 16th IEEE inter. Conference on Networks, 2008.

[14] W.T. Zhu, J. Zhou, R.H. Deng, F. Bao, "Detecting node Replication attacks in wireless sensor networks: A Survey," Journal of Network and Computer Applications: 1022-1034, 2012