SYMMETRIC KEY MANAGEMENT SCHEME FOR HIERARCHICAL WIRELESS SENSOR NETWORKS

Mohammed A. Al-taha[1] and Ra'ad A. Muhajjar[2]

[1]Department of Computer Science, College of Science, Basrah University, Iraq

[2]Department of Computer Science, College of Computer Science and Information Technology, Basrah University, Iraq

ABSTRACT

Wireless Sensor Networks (WSNs) are critical component in many applications that used for data collection. Since sensors have limited resource, Wireless Sensor Networks are more vulnerable to attacks than other wireless networks. It is necessary to design a powerful key management scheme for WSNs and take in consideration the limited characteristics of sensors. To achieve security of communicated data in the network and to extend the WSNs lifetime; this paper proposes a new scheme called Symmetric Key Management Scheme (SKMS). SKMS used Symmetric Key Cryptography that depends only on a Hash function and XOR operation for securing homogeneous and heterogeneous hierarchical WSNs. Symmetric Key Cryptography is less computation than Asymmetric Key Cryptography. Simulation results show that the proposed scheme provides security, save the energy of sensors with low computation overhead.

KEYWORDS

Wireless sensor Networks, Key Management, Symmetric Cryptography, hash function, XOR

1. INTRODUCTION

Wireless sensor networks (WSNs) consist of hundreds, even thousands of low-cost devices called sensor nodes. These sensors have resource constraints such as limited power resource and low memory. Sensors are randomly deployed in open and harsh environments to collect different types of data such as pressure, temperature, soil makeup, humidity, noise levels etc. Sensors in WSNs can communicate with each other by radio channel to transmit the data to the centric node known as the sink node (Base Station). WSNs has formed the basics for covering a different range of applications such as health care, military, environmental monitoring and other fields [1].

WSNs can be classified as flat networks and hierarchical networks. In flat networks, every sensor in the network has the same characteristics (battery lifetime, storage capacity, processor and transmitted power) and perform the same task. In flat networks sensor nodes can transmit data to its neighbor one by one to the sink. In hierarchical networks, the network divided in to several groups called clusters (each cluster has a head called cluster head and the other nodes called cluster members). Hierarchical WSNs can be implemented as homogeneous and heterogeneous. All sensor nodes have the same capabilities in homogeneous WSNs. In heterogeneous WSNs incorporate different types of sensor nodes that have different capabilities (small number of sensors with powerful characteristics elects as Cluster

DOI: 10.5121/ijnsa.2018.10302

head and a large number of low characteristics sensors elects as Cluster members).Most applications that use WSNs [2] have sensitive data as in military applications. Due to limitation resource of sensor node and the hostile environment make it a big challenge to secure the network. Key management is the first requirements to secure communication in WSNs. Key management includes generation, distribution and installation the keys inside sensors and it should support the node addition and deletion with revocation and update the keys [3].Symmetric and Asymmetric cryptography are being used to achieve security (authentication, integrity, privacy) in WSNs. In Symmetric, each node (sender and receiver) shared the same secret key that used for encryption and decryption the data communicated in the network. In Asymmetric key, each node, the sender and receiver, have two keys which are the public key that known to all nodes in the network and secret key which is private [5]. Symmetric cryptography is less computation and consuming less energy than Asymmetric Key [4].

In this paper, Symmetric Key Management Scheme (SKMS) for securing heterogeneous and homogeneous hierarchical WSNs has been proposed. SKMS used symmetric key cryptography depends only on a hash function and XOR operation to establish a session keys between any two nodes.

2. RELATED WORKS

Many key management schemes have been proposed to protect the communication among sensors in the WSNs. In [6] proposed a hierarchical key management scheme (HKMS) to enhance the security hierarchical WSNs. In this scheme, the keys are established and distributed according to the hop count by using one way function. To protect the network from the compromised nodes, different keys were generated based on the hop counts. After particular period of time, the used keys were changed over time.. The authors in [7], used Hybrid key management for heterogeneous WSN, they used Symmetric and Asymmetric cryptographic technique in three levels, first level used signature encryption algorithm based on Elliptic Curve Cryptography (ECC) to secure the communication between the sink and the cluster head. In second level, Diffie-Hellman key exchange algorithm based on ECC is used to generate shared key between the cluster head and sensor nodes. finally, Symmetric session key is used between two sensor nodes because the limited resource of the sensor nodes. Each cluster head selects random value r and broadcast it to all cluster members.

The scheme used shared value based on r when two nodes want to communicated. The authors of [8] proposed a key management scheme for hierarchical WSN by using Power aware routing protocol and track – sector clustering, the track sector clustering scheme is used for minimizing the transmission data redundant by means of reducing the connection between the sink node and the cluster head. Hybrid Elliptic Curve Cryptography (HECC) technique is implementing that used 80 bits key size for securing the routing. In [9] the authors proposed a new scheme used three types of keys: Network key, group key and pairwise key. The network key is used for encrypting the broadcast message and for authenticate the new node, group key is used as a shared key between all the sensor nodes in the same cluster, and the last key, the pairwise key is shared between specific pair of nodes. In the proposed scheme, they used assistant node to improve the security and reduce resource cost when cluster head is compromised. In [10], The authors proposed key management for hierarchical WSN using UAV to establish session key between two nodes. The proposed method used symmetric cryptographic key management which is depended on XOR operation and hash function. UAV used as center unit to reduce the storage. Two nodes can’t establish session key only with the help of UAV. In [11], proposed a hierarchical key management scheme for heterogeneous WSNs that used symmetric cryptographic algorithms with low cost operations such as modular multiplication and bitwise

XOR operation. This scheme has two parts: key generation scheme and key updating scheme. The first part used four types of keys: individual key, pairwise key, cluster key and master key. Individual key used to secure the communication between sensor node and base station. Pairwise key is use to secure the communication between two neighbor nodes. Cluster key is used to secure the communication between all cluster member. Master key is shared with all senor node in the network which use when the base station securely broadcast information to all sensor node in the network. The update part used to update the cluster key and the master.

3. The Proposed Method

3.1 Network Model

The proposed scheme adopts the hierarchical WSN as shown in the Figure 1.

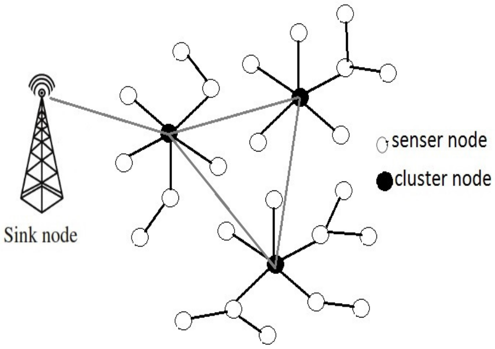


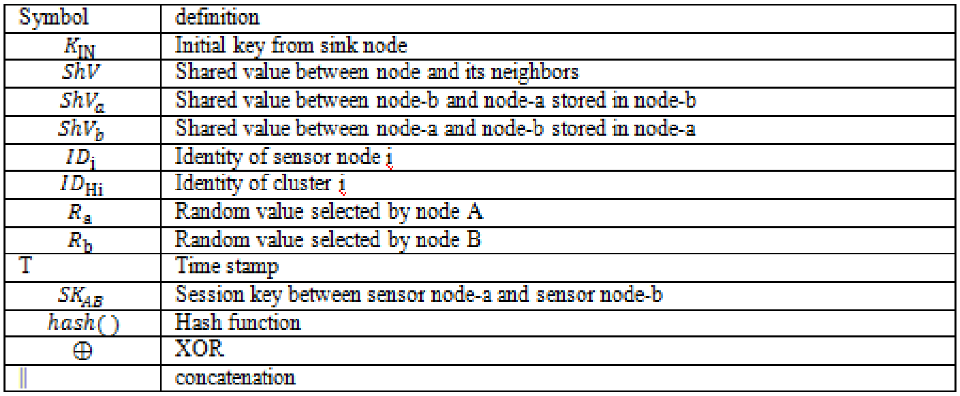
Figure 1. Network Model

The proposed scheme assumes that network has the following properties:

1- The sink node has unlimited storage and sufficient energy with trusted place.
2- All sensor nodes are static.
3- If sensor node is captured by attacker, sensors can extract all information from it.

The notation of our scheme is summarized in table 1:

Table 1. Notation



Symbol	definition
K_{IN}	Initial key from sink node
ShV	Shared value between node and its neighbors
ShV_a	Shared value between node-b and node-a stored in node-b
ShV_b	Shared value between node-a and node-b stored in node-a
ID_i	Identity of sensor node i
ID_{Hi}	Identity of cluster i
R_a	Random value selected by node A
R_b	Random value selected by node B
T	Time stamp
SK_{AB}	Session key between sensor node-a and sensor node-b
$hash()$	Hash function
$\oplus$	XOR
||	concatenation

3.2 Pre-Distribution Phase

Before deployment of the sensor nodes, sink node selected initial key K_{IN}. Each node is assigned a unique ID_{i} and pre-loaded with the initial K_{IN}

3.3 Key Management Phase

After the deployment, each cluster head broadcasts HELLO message which include its ID. Each sensor node may receive messages that coming from more than one cluster head. Sensor node chose a cluster head that have best signal strength. Sensors are storing one more parameter which is the cluster head identity. Now each sensor node has three parameters: initial key, its identity and cluster head identity.

3.3.1 Discovery Phase

In this phase, each node will discover its neighbors and calculates shared value ShV for each neighbor node that is differ from other neighbors and will use this value to establish session keys. The discovery phase can be described as follows:

1- Node A select nonce random value R_a and compute:

$$X_a = R_a \oplus K_{IN}$$
$$Y_a = hash\,(R_a \,||\, K_{IN} \,||\, T_1)$$

Then node A send [X_a, Y_a, T_1] to node B

2- Node B receive [X_a, Y_a, T_1] from node A, first it verified the time stamp whether $| T_1 - T_C | < \Delta T$ or not. If verification holds then computes

$$R_a' = X_a \oplus K_{IN}$$
$$Y_a' = hash\left(R_a' \,||\, K_{IN} \,||\, T_1\right)$$

If $Y_a' = Y_a$ then node B select a random nonce R_b , otherwise send a rejection message to node A.

Now node B computes:

$$X_b = R_b \oplus K_{IN}$$
$$Y_b = hash\,(R_b \,||\, K_{IN} \,||\, T_2)$$

Then node B send [X_b, Y_b, T_2] to node A

3- Node A receive [X_b, Y_b, T_2] from node B, first it verified the time stamp whether $| T_2 - T_C | < \Delta T$ or not. If verification holds then computes

$$R_b' = X_b \oplus K_{IN}$$
$$Y_b' = hash\left(R_b' \,||\, K_{IN} \,||\, T_2\right)$$

If $Y_b' = Y_b$ proceed further, otherwise send a rejection message to node B.

4- Finally, both node A and B agree on same shared value between them

$$ShV = hash\,(R_A \oplus R_B)$$

3.3.2 Key Establishment Phase

Suppose that two sensor nodes A and B are neighbors and want to be communicated, session key process is presented as bellow:

1- First node A send its ID to node B

2- Node B select nonce random value R_B and compute:

$$X_B = R_B \oplus ShV_a$$
$$Y_B = hash\,(R_B \,||\, ShV_a \,||\, T_1)$$

Then node B send [X_B , Y_B , T_1] to node A

3- Node A receive [X_B , Y_B , T_1] from node B, first it verified the time stamp whether $| T_1 - T_C | < \Delta T$ or not. If verification holds then computes

$$R_B^{'} = X_B \oplus ShV_b$$
$$Y_B^{'} = hash\left(R_B^{'} \,||\, ShV_b \,||\, T_1\right)$$

If $Y_B^{'} = Y_B$ then node A select a random nonce R_A , otherwise send a rejection message to node B.

Now node A computes:

$$X_A = R_A \oplus ShV_b$$
$$Y_A = hash\,(R_A \,||ShV_b \,||\, T_2)$$

Then node A send [X_A , Y_A , T_2] to node B

4- Node B receive [X_A , Y_A , T_2] from node A, first it verified the time stamp whether $| T_2 - T_C | < \Delta T$ or not. If verification holds then computes

$$R_A^{'} = X_A \oplus ShV_a$$
$$Y_A^{'} = hash\left(R_A^{'} \,||\, ShV_a \,||\, T_2\right)$$

If $Y_A^{'} = Y_A$ proceed further, otherwise send a rejection message to node A.

5- Finally, both node A and B agree on same session key

$$SK_{AB} = hash\,(R_A \oplus R_B)$$

4. Security Analysis

The security analysis of the proposed scheme can be discussed as following:

4.1 Key updating

In the proposed scheme, each session used key different from the others. Every session key depends on shared value between two nodes. For more security, the shared value between two nodes should be updated periodically. After certain period, the update phase is started.

First, each node updates the initial key by tack hash value for the current initial key.

$$K_{IN}' = hash\,(K_{IN})$$

After that each node update the shared value with its neighbors by using the new initial key K_{IN}' as follows:

1- Node A select nonce random value R_a and compute:

$$X_a = R_a \oplus K_{IN}'$$
$$Y_a = hash\,(R_a \,||\, K_{IN}' \,||\, T_1)$$

Then node A send [X_a , Y_a , T_1] to node B

2- Node B receive [X_a , Y_a , T_1] from node A, first it verified the time stamp whether $| T_1 - T_C | < \Delta T$ or not. If verification holds then computes

$$R_a^{'} = X_a \oplus K_{IN}'$$
$$Y_a^{'} = hash\left(R_a^{'} \,||\, K_{IN}' \,||\, T_1\right)$$

If $Y_a^{'} = Y_a$ then node B select a random nonce R_b , otherwise send a rejection message to node A.

Now node B computes:

$$X_b = R_b \oplus K_{IN}'$$

$$Y_b = hash\,(R_b \,||\, K_{IN}' \,||\, T_2)$$

Then node B send [X_b, Y_b, T_2] to node A

3- Node A receive [X_b, Y_b, T_2] from node B, first it verified the time stamp whether $| T_2 - T_C | < \Delta T$ or not. If verification holds then computes

$$R_b^{'} = X_b \oplus K_{IN}'$$

$$Y_b^{'} = hash\left(R_b^{'} \,||\, K_{IN}' \,||\, T_2\right)$$

If $Y_b^{'} = Y_b$ proceed further, otherwise send a rejection message to node B.

4- Finally, both node A and B agree on same shared value between them

$$ShV' = hash\,(R_A \oplus R_B)$$

4.2 ADD NEW NODES

The sensor nodes have limited energy and after some time the node will die, for that, the died node should be replaced with new node. For more security, after period each node update the initial key by take hash value of the current key as follows:

$$K_{IN}' = hash\,(K_{IN})$$

Before deployment the new node, the sink node is known the current value of the initial key. So, sink node pre-loaded the new node with its ID and the current value of initial key K_{IN} . After deployment the new node, first the node joined a cluster and it start the discovery phase with its neighbors to calculate the shared values with them and the neighbor nodes continued with the update phase.

4.3 KEY REVOCATION

When a node captured, all the security information stored on it will become compromised. Each node that have shared value with the compromised node will delete it and then the nodes in the same cluster that compromised node belong to it will enter the discovery phase and calculate new shared value among them.

5. PERFORMANCE EVALUATION

In this section, we evaluate the proposed scheme and analyzed the storage overhead, energy consumption and time which compared with other related work. The proposed scheme compared with scheme *"Lightweight Multilevel Key Management Scheme for Large Scale Wireless Sensor Network"* (MKMLS) [10] under the same circumstance. In our simulation environment, we used 500 sensors that randomly deploy with area 200m*200m, 25 sensors as cluster head and 475 as sensor nodes. The transmission range of cluster head is 40m and the sensor nodes is 20m with initial energy 5J and 15J respectively.

5.1 STORAGE OVERHEAD

The key length of the proposed scheme is 128-bit as in MKMLS. In SKMS before the deployment, each sensor node is preloaded with its ID and the initial key, after deployment, each node in the same cluster enter in the discovery phase with its neighbors and calculate shared value with each neighbor that differ from other. This value will use to establish session keys. In MKMLS, before the deployment, each node is preloaded with its ID and a secret key. After deployment, each sensor node need to store one more shared value α12. The total number of this value is fixed by its neighbors, i.e. each sensor node need to store a shared value with each neighbor in its memory and will used it when they want to establish session keys. Compared with MKMLS scheme, SKMS have the same storage overhead.

5.2 Energy Consumption

The energy consumption of our SKMS to send 1000 packets is less from the MKMLS scheme; figure 2 shows the energy consumption of sending 1000 packets of both SKMS and MKMLS

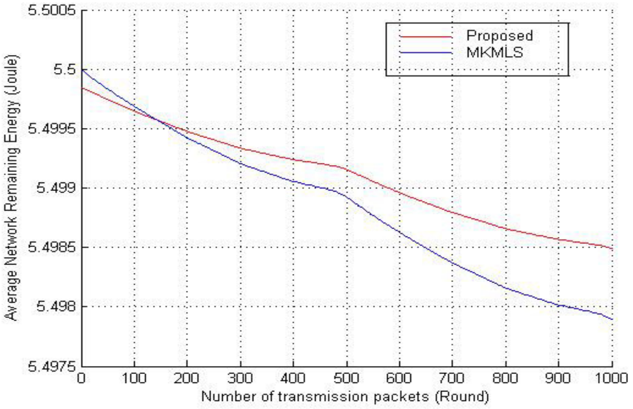


Figure 2. Energy Consumption for transmit 1000 packets

5.3 Time Consumption

The proposed scheme takes less time to generate session keys and transmit the sensed data. Figure 3 shows the time for transmit 1000 packets and compared with MKMLS.

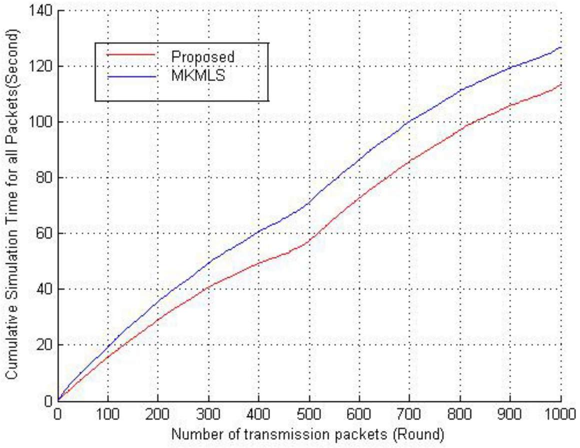


Figure 3. Time Consumption for transmit 1000 packets

6. Conclusion

In this paper, a Symmetric Key Management Scheme (SKMS) for hierarchical Wireless Sensor Networks was presented. The proposed scheme used symmetric cryptography depends only on a hash function and XOR operation to establish a session keys between any two nodes. Through performance evaluation, we extend the network lifetime by consuming less energy and take less time to establish secure communication among the nodes.

SKMS use shared value between nodes to establish symmetric session keys and update that value at regular interval to avoid node capture attack and to assure that only legal nodes can be communicated. SKMS is suitable for different types of WSNs, it works in both flat and hierarchical WSNs and for homogeneous and heterogeneous WSNs. The shared value between nodes is calculated after the deployment in the discovery phase which requires three steps and the key establishment phase needs five steps to generate the session key and because of that SKMS take less time to calculate the shared value and to generate the session keys. However, in MKMLS, it used UAV as a key management center. If two nodes want to be communicated for the first time, the two nodes cannot establish session key only with the help of UAV. MKMLS needs eight steps just to calculate the shared value between nodes and five more for establish the session key. Each step requires negotiation between nodes and each step consuming time and energy. Comparing with MKMLS, SKMS needs fewer steps to calculate the shared value and to establish the session key and as a result, consuming less time and consuming less energy.

Simulation and analysis shows that SKMS has good energy efficient, less time consuming than other similar schemes

REFRENCES

[1] Ahmed A. Alkadhmawee and Songfeng Lu, "Prolonging the Network Lifetime Based on LPA-Star Algorithm and Fuzzy Logic in Wireless Sensor Network," World Congress on Intelligent Control and Automation (WCICA), IEEE, 2016

[2] Pawgasame, W., "A survey in adaptive hybrid wireless Sensor Network for military operations". IEEE, in Second Asian Conference Defence Technology (ACDT), pp. 78-83, 2016

[3] J. Zhang and V. Varadharajan, "Wireless sensor network key management survey and taxonomy", Journal of Network and Computer Applications, vol. 33, pp. 63-75, 2010.

[4] Omer K. Jasim, Safia Abbas and El-Sayed M. Horbaty, "Evolution of an Emerging Symmetric Quantum Cryptographic Algorithm", Journal of Information Security, Vol. 6, pp. 82-92, 2015

[5] Ayman T., Ayman K. and Ali C., "Authentication Schemes for Wireless Sensor Networks," in Mediterranean Electrotechnical Conference (MELECON), IEEE, pp. 367-372, 2014

[6] Y. Zhang, X. Li, J. Liu, J. Yang, and B. Cui, "A secure hierarchical key management scheme in wireless sensor network," International Journal of Distributed Sensor Networks, 2012.

[7] Ying, Z. and Pengfei, J. "An Efficient and Hybrid Key Management for Heterogeneous Wireless Sensor Networks" ,Control And Decision Conference (2014 CCDC),The 26th Chinese , ,IEEE, 2014

[8] V Vijayalakshmi, R Sharmila, and R Shalini. "Hierarchical key management scheme using hyper elliptic curve cryptography in wireless sensor networks". In Signal Processing, Communication and Networking (ICSCN), 2015 3rd International Conference on. IEEE, 2015

[9] Zhang, X., and Wsn, A. C. "An Efficient Key Management Scheme in Hierarchical Wireless Sensor Networks". 2015 International Conference on Computing, Communication And Security (ICCCS), IEEE,doi:10.1109/CCCS.2015.7374122, 2015

[10] Akansha Singh, Amit K. Awasthi and Karan Singh," Lightweight Multilevel KeyManagement Scheme for Large Scale Wireless Sensor Network",IEEE 2016

[11] C. M. Chen, X. Zheng, and T. Y. Wu, "A Complete Hierarchical Key Management Scheme for Heterogeneous Wireless Sensor Networks," The Scientific World Journal, vol. 2014, article ID 816549, 2014.