ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 6, July 2017

# Localized Encryption and Authentication Protocol for Secure Key Management in Wireless Sensor Networks

Arundhati Nelli [1], Sushant Mangasuli [2] Manasa N [3]

Assistant Professor, Dept. of CSE, Alva's Institute of Engineering and Technology, Mijar, Karnataka, India [1]

Assistant Professor, Dept. of CSE, Alva's Institute of Engineering and Technology, Mijar, Karnataka, India [2]

P.G Student (M. Tech.), Dept. of CSE, Alva's Institute of Engineering and Technology, Mijar, Karnataka, India [3]

**ABSTRACT**: Wireless Sensor Networks (WSNs) are spread widely and rapidly due its unique features, such as their light weight low-cost, and adhoc nature. However, this network is vulnerable to several attacks that affect its security. The security of WSN has very crucial issue nowadays in transmitting data over network. Due to the resource limitations of sensor nodes, providing security protocols is a particular challenge in sensor networks. A popular proposed method is Localized Encryption and Authentication Protocol (LEAP), is an efficient and light-weight protocol, but includes loopholes through which adversaries may launch replay attack by successfully masquerading as legitimate nodes and thereby compromise the communications over the network. LEAP supports the establishment of four types of keys. The security of these keys is under the assumption that the initial deployment phase is secure and the initial key is erased from sensor nodes after the initialization phase. However, the initial key is used again for node addition after the initialization phase whereas the new node can be compromised before erasing the key. A time based key management scheme rethought the security of LEAP. This paper gives brief description about strength and weaknesses of LEAP.

**KEYWORDS**: WSN, LEAP, Security, Key Management

## I. INTRODUCTION

Wireless sensor networks (WSNs) are distributed systems consisting of a large number of sensor nodes and a base station as a controller which interface the sensor network to the outside network. WSNs may be deployed in unattended and adversarial environments such as battlefields. Compared to conventional networks, they are more vulnerable to physical destruction and man-made threats. Therefore, providing security is a particular challenge in sensor networks due to the resource limitations of sensor nodes, wireless communications and other related concerns. As a specific example, it is impractical to use asymmetric cryptosystems in sensor networks in which each node has low operational capability and insufficient memory. In a WSN , various types of communication may happen. The base station broadcasts control commands to the whole network. Control node multicasts messages within the cluster. A node communicates with its neighboring nodes by unicasting. Therefore, network-wide key, cluster key, and pairwise key are required to satisfy different types of secure communication. Therefore Adevised a scheme called localized encryption and authentication protocol (LEAP) for WSNs. LEAP (Localized Encryption and Authentication Protocol) is a protocol with key management scheme that is very efficient with its security mechanisms used for large scale distributed sensor networks. It generally supports for inside network processing such as data aggregation. In-network processing results in reduction of the energy consumption in network. To provide the confidentiality and authentication to the data packet, LEAP uses multiple keys mechanism. For each node four keys are used known as individual, pair wise, cluster and group key. All are symmetric keys and use as follows:
**Individual Key:** It is the unique key used for the communication between source node and the sink node.
**Pairwise Key:** It is shared with another sensor nodes.

**Cluster Key:** It is used for locally broadcast messages and shares it between the node and all its surrounding neighboring nodes.

**Group Key:** globally shared key used by all the network Nodes.

These keys can also be used by other non-secured protocols to increase the network security. LEAP is satisfies several security and performance requirements of WSN.

The security of all types of keys relys on that of the initial key. As many existing key management protocols, LEAP assumes that the initial key is secure during the initialization phase and is erased from the memory of sensor nodes when the initialization phase finishes. However, the same key should be used again for node addition and replacement. Some new nodes may be captured at any time after the initialization phase. That is, the new deployed nodes could be captured before removing the initialization key. The security of the scheme is threatened by the attacks launched after the initialization phase. LEAP also includes an efficient protocol for local broadcast authentication based on the use of one-way key chains. A salient feature of the authentication protocol is that it supports source authentication without precluding in-network processing. LEAP is very efficient in computation, communication, and storage. We analyze the security of LEAP under various attack models and show that LEAP is very effective in defending against many sophisticated attacks such as HELLO Flood attack, Sybil attack, and Wormhole attack.

## II. The Need For Key Management

Before a WSN can exchange data securely, encryption keys must be established among sensor nodes. Key distribution refers to the distribution of multiple keys among the sensor nodes, which is typical in a non-trivial security scheme. Key management is a broader term for key distribution, which also includes the processes of key setup, the initial distribution of keys, and key revocation — the removal of a compromised key. Key management, like security, is a cross-layered issue. The need for key management starts in the link layer. An applicable link layer standard in a WSN is IEEE 802.15.4. Although this standard considers key usage for secure data transmission, it does not specify how to exchange keys securely. This leaves open the key management problem that is the focus of much recent research. Besides the link layer, upper layers such as the network and application layers also must exchange keys securely. Many security-critical applications depend on key management processes to operate but also demand a high level of fault tolerance when a node is compromised. This is a challenging problem because there are many stringent requirements for key management, and the resources available to implement such processes are highly constrained.

## III. Security and operational requirement for key management

Key management requirements can be divided into security requirements that form a subset of the overall WSN security requirements and operational requirements that act as constraints in the design and realization of key management.

**SECURITY REQUIREMENTS:**

**Data Confidentiality:** In sensor network, data flows from many intermediate nodes and chance of data leak is more. To provide the data confidentiality, an encrypted data is used so that only recipient decrypts the data to its original form.

**Data Integrity:** Data received by the receiver should not be altered or modified is Data Integrity. Original data is changed by intruder or due to harsh environment. The intruder may change the data according to its need and sends this new data to the receiver.

**Data Authentication:** It is the procedure of confirmation that the communicating node is the one that it claims to be. It is important for receiver node to do verification that the data is received from an authenticate node.

**Data Availability:** Data Availability means that the services are available all the time even in case of some attacks such as Denial of service.

**Source Localization:** For data transmission some applications use location information of the sink node. It is important to give security to the location information. Non-secured data can be controlled by the malicious node by sending false signal strengths or replaying signals.
**Self-Organization:** In WSN no fixed infrastructure exists, hence, every node is independent having properties of adaptation to the different situations and maintains self organizing and self healing properties. This is a great challenge for security in WSN.
**Data Freshness:** Data freshness means that each message transmitted over the channel is new and fresh. It guarantees that the old messages cannot be replayed by any node. This can be solved by adding some time related counter to check the freshness of the data.
**Scalability:** It should sustain a big number of nodes.

**OPERATIONAL REQUIREMENTS:**
**Accessibility:** Intermediate nodes should be able to perform data aggregation by combining data from different nodes. Neighboring nodes should also be able to passively monitor event signals to prevent large amounts of redundant event signaling information.
**Flexibility**: Nodes should be replaceable when compromised. On-the-fly addition of nodes should also be supported.
**Scalability:** A WSN should concurrently support at least 3000 nodes even with the key management scheme in place.

## IV. KEY DISTRIBUTION SCHEMES

The three simplest keying models that are used to compare the different relationships between the WSN security and operational requirements are network keying, pairwise keying, and group keying. A detailed problem and benefit analysis is given in below table.

| Model | Description | Benefits | Problems |
|---|---|---|---|
| Network | The entire network uses one shared secret key. | Simple<br>Allows data aggregation and fusion Scalable<br>Able to self-organize<br>Flexible/accessible | Compromise of one node compromises the entire Network |
| Pairwise | Each specific pair of nodes shares a different key. | Best robustness<br>Authenticates each node | Nonscalable --storage, energy, computation.<br>Unable to self-organize |
| Group | Each group uses a different shared key. | Allows multicast<br>Allows group collaboration<br>Better robustness than network-wide keying<br>Adjustable scalability | Lacks efficient storage method for group keying in IEEE 802.15.4.<br>Difficult to set up securely |

(a) (d) (e) (c)

## V. KEY MANAGEMENT PROTOCOL

To realize a practical, robust keying model, in this paper efficient key management protocol that address the problems in each of the three basic schemes discussed previously. The localized encryption and authentication protocol (LEAP), which employs a hybrid approach. This is a *jack-of-all-trades* protocol offering network-wide, cluster/group, and pairwise keying capabilities. To accomplish this, LEAP uses four types of keys: *individual*, *group*, *cluster*, and *pairwise shared keys*. The *individual key* is unique for each sensor node to communicate with the sink node. The Individual key is a shared between a node and its corresponding base station in order to provide security between them as they communicate. Communication between a node and a base station is vital as it allows a node to inform the base station of any abnormal behavior detected from its surrounding nodes. As a result, the base station being aware of the malicious node can then use the key to encrypt the important information such as instructions to a specific node. The *group key* is a network-wide key for communication from the sink node to all sensor nodes. A group key, also known as the global key is shared by all the sensor nodes within the network. The base station uses this key to encrypt data that is transmitted to all the nodes within the group. Since the entire group of nodes is sharing this key, it eliminates the need for a base station to separately encrypt the same message to individual nodes with individual keys. Confidentiality is invoked as long as the key is updated every once a while in case one of the nodes stops functioning and is removed from the group or network. A special case of a group key is known as the cluster key. An authentication mechanism known as μTimed Efficient Streaming Loss-tolerant Authentication Protocol (μ*TESLA*) can also be used for the broadcast authentication of the sink node, which ensure that packets sent with the group key are from the sink node only.

The *cluster key* is used for collaborations within a cluster. The key is shared by a node with multiple of its neighboring sensor nodes. The cluster key is generated by node using a random function and encrypts this key using the pairwise key so that only the authenticated neighbors are able to decrypt to get access to the cluster key An authentication mechanism known as a one-way hash-key chain that employs a non-reversible mathematical operation is used to ensure that the source of the packet can be authenticated without precluding passive data aggregation. Lastly, the *pairwise shared key* is used for secure communications between neighboring nodes. Key shared between a node and its neighbouring sensor nodes. The establishment of this key ensures protection of communication that longs for privacy or authentication of a source. The advantage of having a pair-wise key secures transmission because it is shared between a node and one of its immediate neighbors and therefore prevents it from intruders.

LEAP uses a pre-distribution key to help establish the four types of keys. The individual key is first established using a function of a seed and the ID of the node. Then, in the pairwise shared key phase, a neighbor discovery process is initiated, and nodes broadcast their IDs. The receiving node uses a function, seeded with an initial key, to calculate the shared key between it and all of its neighbors. Afterwards, the initial key and any intermediate keys that were generated are erased. Thirdly, the cluster key is distributed by the cluster head using pairwise communication secured with the pairwise shared key. Lastly, for distributing the network-wide group key, the sink node broadcasts it in a multihop, cluster-by-cluster manner starting with the closest cluster. LEAP has many advantages that satisfy the requirements of WSNs. First, it has μ*TESLA* and one-way key chain authentication, as well as key revocation and key refreshing. The accessibility requirement also can be easily satisfied by encrypting data that requires aggregation with the cluster key. The fine granularity it supports enables data to be encrypted at the correct level (i.e., key level) to ensure reasonable security is achieved without prohibiting data fusion or aggregation. The scalability of LEAP can be analyzed in terms of computational cost and storage cost. The computational cost of LEAP is inversely proportional to the number of nodes and directly proportional to the number of neighbors (i.e., node density because the higher the density of the network, the more connections are formed per cluster. The storage cost also is quite reasonable as pairwise keying is used only for one-hop neighbors. It is apparent that LEAP satisfies both the security and operational requirements very well. The only drawback with LEAP is that it assumes the sink node is never compromised.

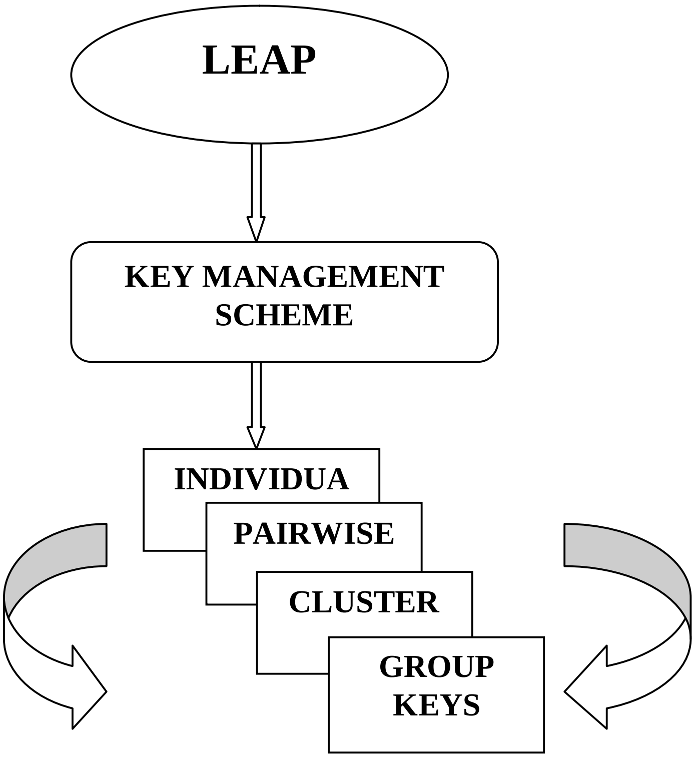

FIG 1: KEY MANAGEMENT PROTOCOL

## VI. CONCLUSION

This paper describes Localized Encryption and Authentication Protocol, a key management protocol for sensor networks.

LEAP has the following properties:

• LEAP includes support for establishing four types of keys per sensor node – individual keys shared with the base station, pairwise keys shared with individual neighboring nodes, cluster keys shared with a set of neighbors, and a group key shared with all the nodes in the network. These keys can be used to increase the security of many non-secure protocols.LEAP includes an efficient protocol for local broadcast authentication based on the use of one-way key chains.

• A distinguishing feature of LEAP is that its key sharing approach supports in-network processing, while restricting the security impact of a node compromise to the immediate network neighborhood of the compromised node.

• LEAP can prevent or increase the difficulty of launching many security attacks on sensor networks.

• The key establishment and key updating procedures used by LEAP are efficient and the storage requirements per node are small.

## REFERENCES

[1] Johnson C. Lee And Victor C. M. Leung, Kirk H. Wong, Jiannong Cao, And Henry C. B. Chan, “Key Management Issues in Wireless Sensor Networks: Current Proposals And Future Developments”, IEEE Access, pp, 1536-1284,2007.

[2] Manel Boujelben, Habib Youssef, Mohamed Abid, “An efficient scheme for key pre-distribution in wireless sensor networks”,IEEE Access, 978-0-7695,2008..

[3] Di Zhang, Yi Zhao, Xingming Wang, Jaeho Choi" A Robust and Efficient Neighborhood-Based Security Protocol for Wireless Sensor Networks",IEEE Access, 978-0-7695-4235-5,2010.

[4] C. Castelluccia, and A. Spognardi, RoK: "A Robust Key Predistribution Protocol for Multi-phase Wireless Sensor Networks", Proceedings of SecureComm 200 73rd International Conference on Security and Privacy in Communications Networks, pp.351-360, 2007.

[5] K. Kalkan, S. Yilmaz, O. Z. Yilmaz, A. Levi, "A Highly Resilient and Zone-based Key Predistribution Protocol for Multiphase Wireless Sensor Networks", International Symposium on QoS and Security for Wireless and Mobile Networks, Tenerife, Spain, pp.2936. 2009.

[6] R. Verma and B. Basile. "Modeling and analysis of leap, a key management protocol for wireless sensor networks."Technical Report UHCD-08-12, Department of Computer Science, University of Houston, August 2008. G. Barrenetxea,

[7] "Distributed routing algorithms for sensor networks", PhD thesis at Ecoles Polytechniques f´ed´erales, Switzerland, 2006.