International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

Improved Key Management and Security in Dynamic Wireless Sensor Networks

Kiran .E[1], T. Nivethitha[2]

[1]PG Scholar, M.E Applied Electronics, Rathinam Technical Campus, Coimbatore

[2]Assistant Professor, ECE Department, Rathinam Technical Campus, Coimbatore

Abstract: ***Wireless sensor networks (WSNs) have been a great hand for many applications, military communication and data handling, monitoring of patient from different locations, the common traffic system is made efficient and traffic flow monitoring are a few among the applications. Suitable encryption methods are required for security and communication. A certificate less-effective key management (CL-EKM) protocol provides a secure communication in dynamic wireless sensor networks. Efficient key update when a node joins or leaves a cluster is supported in CL-EKM. Also provide forward and backward secrecy in node to node communication. Efficient key revocation for compromised nodes helps in minimizing the impact of a node compromise on the security of other nodes or links of communication. A security analysis of this scheme shows that the protocol is effective in resisting various attacks. Implementing CL-EKM in ubuntu OS and simulate it using NS2(NS2.34) simulator to assess its energy, time, communication performance.***

Keywords: certificate less public key cryptography, key management scheme, forward and backward key secrecy, certificate less-effective key management protocol, dynamic wireless sensor networks

1. Introduction

Wireless Sensor Networks are consisting of distributed autonomous sensors to monitor physical or environmental conditions, like the temperature, sound, pressure, etc. and also to cooperatively pass the data through the network to a main location. The modern networks are bi-directional, also we can control the sensor activity.

The development of wireless sensor networks was motivated by the military applications like battlefield surveillance; today, such networks have been used in many industrial and consumer applications, like industrial process monitoring and control, machine health monitoring. The WSN consists of "nodes" from a few to several hundreds or even thousands, where each node connected to one (or sometimes several) sensors. Each such sensor network node will have typically several parts: a radio transceiver which has an internal antenna or connection to an external antenna, a microcontroller, an electronic circuit for the interface with the sensors and an energy source, usually a battery or an embedded form of energy storage. A sensor node may vary in size from that of a shoebox down to the size of a grain of dust, although functioning "motes" of genuine microscopic sized prototype is yet to be created. The costs of sensor nodes are similarly variable, ranging from a few to hundreds of USD; which depends on the complexity of the individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on the available resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can be varied from a simple star network to advanced multi-hop wireless mesh network. The propagation technique between the hops of the network can be by routing or flooding.

Topology control is a technique which is used in distributed computing for altering the underlying network (modeled as a graph) for reducing the cost of distributed algorithms if run over the new resulting graphs. It is one of the basic techniques in distributed algorithms. For instance, a (minimum) spanning tree will be used as a backbone for reducing the cost of broadcast from O(m) to O(n), where n and m are the number of vertices and edges in the graph, respectively. The term "topology control" which is used mostly by the wireless ad hoc and sensor network research community. The main aim of topology control in this domain is for saving energy, reducing interference between nodes and to extend lifetime of the network.

Once the initial topology have been deployed, especially when the locations of the nodes are random, the administrator won't have any control over the design of the network. However, the administrator will have control over some parameters of the network: transmission power of the nodes, state of the nodes (whether it's active or sleeping), role of the nodes (gateway, Cluster head, regular), etc. With the modification of these parameters, the topology of the network can be changed.

Upon the same time a topology will get reduced and the network starts serving its purpose, the selected nodes will start spending energy: Reduced topology will start losing its "optimality" as soon as the full network activity evolves. After some time while it's active, some nodes might start to run out of energy. Especially in WSN with multihopping, intensive packet forwarding causes the nodes that are closer to the sink to spend higher amounts of energy than that of the nodes that are farther away.

Key update protocol (also known as "key establishment") is any method in the cryptography by which cryptographic keys are exchanged between two parties, allowing the use of a cryptographic algorithm. If the sender and the receiver wish to exchange encrypted messages, each should be equipped to encrypt messages that are to be sent and decrypt messages received. The nature of the equipping requirement depends on the encryption technique that is used. If they use a code, both shall require a copy of the same codebook. If they use a cipher, they will need appropriate keys. If the cipher is of a symmetric key cipher, both shall need a copy

of the same key. If an asymmetric key cipher with the public/private key property, both shall need the other's public key.

2. Overview

A certificate less effective key management (CL-EKM) scheme for dynamic WSNs is proposed.

- In certificate less public key cryptography (CL-PKC), the user's full private key is the combination of a partial private key generated by key generation center (KGC) and the user's own secret value.

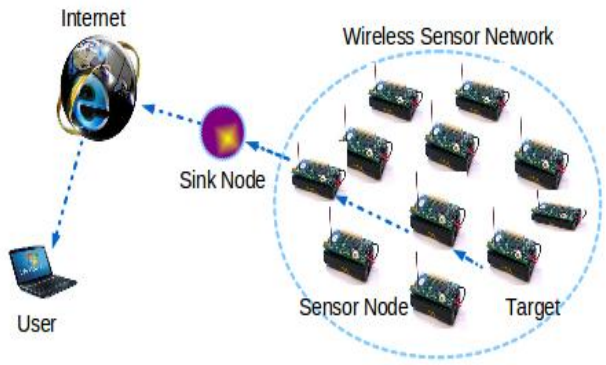

Figure 1: Proposed architecture

- The special organization of the full private/public key pair will remove the need for certificates and also will resolve the key escrow problem by the removal of the responsibility for the user's full private key.
- We can also take the benefit of ECC keys defined on an additive group with 160-bit length as secured as the RSA keys with 1024-bit length.

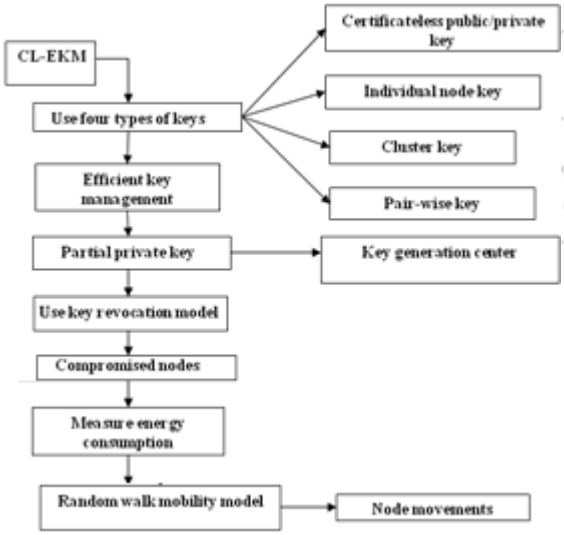

Figure 2: flow diagram of proposed system.

2.1 Types of Keys

- **Certificate less Public/Private Key**: Before a node is deployed, the KGC at the BS generates a unique certificate less private/public key pair and installs the keys in the node. This key pair would use to generate a mutually authenticated pair wise key.
- **Individual Node Key**: Each node will share a unique individual key with BS. For example, a L-sensor can use an individual key to encrypt an alert message that is sent to the BS, or if it shows to be failed to communicate with the H-sensor. An H-sensor will use its individual key to encrypt the message corresponding to the changes in the cluster. The BS will also use the same key to encrypt any sensitive data, such as compromised node information or commands. Before a node is deployed, the BS will assign the node the individual key.
- **Pair wise Key**: Each node shares different pair wise key with each of its neighboring nodes for secure communications and authentication of these nodes. For example, to join a cluster, L-sensor must share a pair wise key with the H-sensor. Then, the H-sensor can be securely encrypt and distribute its cluster key to the L-sensor using the pair wise key. In aggregation supportive WSN, the L-sensor may use its pair wise key to securely transmit the sensed data to the H-sensor. Each node shall dynamically establish the pair wise key between itself and another node by using their respective certificate less public/private key pairs.
- **Cluster Key**: All nodes in a cluster will share a key, named as cluster key. The cluster key is used for securing broadcast messages in a cluster, e.g sensitive commands or change of member status in the cluster. Only the cluster head is able to update the cluster key when a L-sensor leaves or joins the cluster.

3. Implementation

System implementation is a stage in the project where the theoretical designs will be turned into working system. The most crucial stage which improve the user's confidence that the new system will work effectively and efficiently. Proper implementation is an essential thing to provide a reliable system for meeting organization requirements. During the implementation stage a live demons were undertaken and made in front of end-users. Implementation is the stage that of project when the system design is turned into a working system. The stage consists of the following steps.

- Testing the developed program using a sample data.
- Detection and correction of the internal error.
- Testing the system to get the user requirement.
- Feeding real time data and retesting.
- Making necessary change according to the user's description.

3.1 Key Establishment and Update

Pair wise Encryption Key Establishment

• Choose $l_A \in_R Z_{*q}$ and compute $U_A = l_A P$.
• Find $T_A = l_A \cdot h_0(B, R_B, P_B)P_{pub} + l_A \cdot R_B \bmod q$
$K_{AB} = h_1(U_A, T_A, l_A \cdot P_B, B, P_B)$
• Find $h = h_2(U_A, \tau_A, T_A, A, P_A, B, P_B)$
$h\_ = h_3(U_A, \tau_A, T_A, A, P_A, B, P_B)$
$W_A = d_A + l_A \cdot h + x_A \cdot h\_$
∴ τ_A is a random string to give freshness.
• O/pt K_{AB} and $\phi_A = (U_A, W_A)$.
• Find $T_A = d_B \cdot U_A$.

Note: since $d_B = r_B + x \cdot h_0(B, R_B, P_B)$ and
$U_A = l_A P \bmod q$, T_A is found as $T_A = (r_B + x \cdot h_0(B, R_B, P_B)) \cdot l_A P \bmod q = l_A \cdot h_0(B, R_B, P_B) P_{pub} + l_A \cdot R_B \bmod q$,
• Get $h = h_2(U_A, \tau_A, T_A, A, P_A, B, P_B)$ and
$h\_ = h_3(U_A, \tau_A, T_A, A, P_A, B, P_B)$.
• If $W_A \cdot P = R_A + h_0(A, R_A, P_A) \cdot P_{pub} + h \cdot U_A + h\_ \cdot P_A$,
O/p $K_{AB} = h_1(U_A, T_A, x_B \cdot U_A, B, P_B)$. Otherwise,
O/p invalid.

Pair wise Encryption Key Establishment

Once n_A and n_B are set the pair wise master key K_{AB}, they generate a HMAC of K_{AB} and a nonce $r \in_R Z_{*q}$. The HMAC will then be validated by both n_A and n_B. If the validation is successful, the HMAC value will be established as the short-term pair wise encryption key k_{AB}. It is summarized below:

- n_B selects a random nonce $r \in_R Z_{*q}$, computes
- $k_{AB} = HMAC(K_{AB}, r)$ & $C_1 = E_{kAB}(r, A, B)$. Then, n_B sends r & C_1 to n_A.
- When n_A receives r and C_1, it calculates

$k_{AB} = HMAC(K_{AB}, r)$ and decrypts C_1. Then it checks and validates r, A and B and if valid confirms that n_B knows K_{AB} and it can find k_{AB}.

4. Pair wise and Cluster Key Update

Pair wise Key Update: To update a pair wise encryption key, two nodes which are shared the pair wise key perform a Pair wise Encryption Key Establishment process. On the other hand, the pair wise master key is not requiring periodical updates, because will not be directly used to encrypt each session message. As long as the nodes will not be compromised, the pair wise master keys is not able to be exposed. However, whether a pair wise master key is modified or needs to be updated as the BS demands to be, the Pair wise Master Key Establishment process should be executed.

Cluster Key Update: Only cluster head H-sensors may update their cluster key. If a L-sensor try to change the cluster key, the node will be considered a malicious node. The operation for any j th cluster will then be described as follows:

1) n_{Hj} selects $x\_j \in_R Z_{*q}$ and calculates a new cluster key $GK\_j = HMAC(x\_j, H_j)$. n_{Hj} also generate an Update messages including $HMAC(GK\_j, Update)$ & computes $C_6 = E_{GKj}(GK\_j, HMAC(GK\_j, Update))$. Then, n_{Hj} transmits Update and C_6 to its members in the cluster.
2) Each members n_{Li} decrypts C_6 by using the GK_j, verifies $HMAC(GK\_j, Update)$ and updates the cluster key as GK_j. Then, each n_{Li} send the acknowledgement message to n_{Hj}.

Node Discovery and Authentication
$n_{H_j} \rightarrow *$: $\langle H_j, pk_{H_j} \rangle$
(for $i = 1, \ldots, n$)
$n_{L_i} \leftrightarrow n_{H_j}$: Perform *Pairwise Key Generation* phase

Cluster Key Generation
(for $i = 1, \ldots, n$)
n_{H_j} : Generate GK_j, Compute $C_2 = E_{k_{L_i H_j}}(GK_j, H_j, L_i)$
$n_{H_j} \rightarrow n_{L_i}$: $\langle H_j, C_2 \rangle$
n_{L_i} : Decrypt C_2 to get GK_j and
Compute $C_3 = E_{k_{L_i H_j}}(L_i, HMAC(k_{L_i H_j}, GK_j))$
$n_{L_i} \rightarrow n_{H_j}$: $\langle L_i, C_3 \rangle$
n_{H_j} : Decrypt C_3 and Check the validity

Membership Validation
n_{H_j} : Compute $C_4 = E_{K^0_{H_j}}(H_j, \mathfrak{M}_j)$, $C_5 = E_{GK_j}(H_j, \mathfrak{M}_j)$
$n_{H_j} \rightarrow$ BS : $\langle H_j, C_4 \rangle$
BS : Check $\mathfrak{M}_j$
BS $\rightarrow n_{H_j}$: $\langle Acknowledgement \rangle$
$n_{H_j} \rightarrow *$: $\langle C_5 \rangle$

Figure 3: Cluster Formation algorithm

3.2 Modules

Nodes Deployed And Create Sensor Nodes And Sink Node

- Nodes deployed in the network by the help of ns2
- Create sensor nodes and Sink node of higher efficiency.

Two-Layered Key Management Scheme In Network

- Two layered keys generated for two nodes in a network.
- For dynamically providing both node authentication and establish the pair wise key between nodes, using a pairing-free certificate less hybrid sign encryption scheme
- The adversary will also populate the network with the clones of the captured node.
- Even without capturing a single node, an adversary is able to conduct an impersonation attack by injecting an illegitimate node, which try to impersonate a legitimate node.

Elliptic Curve Cryptography Based Certificate in Sink Communication

- The scheme should assure forward secrecy to prevent a node the chance using an old key to continue decrypting new messages. It should also assure backward secrecy for preventing a node.
- Each node will share different pair wise key with each of their neighboring nodes for the secure communications and authentication of all these nodes.

Formation of Cluster Based On Node Mobility and Location

- A new node to the existing networks, the BS must ensure that the node is not compromised. The new node which is added establishes a full private/public key by the node registration.
- In CL-EKM, messages are exchanged between nodes or within a cluster will be encrypted with the pair wise encryption key or cluster key.
- The cluster head will validate each node with the BS in the node joining process of the CL-EKM, the BS can

detect a cloned node when it is placed in an unintended cluster.

Certificate less Effective Key Management

- The BS or Sink is able to consider a node as compromised if the node is disappearing for a certain period of time. In this case, the BS has to investigate the suspicious node and it can use the node fault detection.
- The BS can discover a compromised node or a compromised cluster head.
- To establish the pair wise encryption of a random Key, other than generating a legitimate master key.

5. Performance Analysis

Network throughput can be defined as the rate of successful message delivery in a communication channel. The data these messages belong to will be delivered over a physical or logical link or it can be passed through a certain network node. Throughput can be usually measured in bits per second (bit/s or bps), or in data packets per second (p/s or pps) or data packets per time slot.

The **system throughput** or **aggregate throughput** may also be termed as the sum of the data rates that delivered to all terminals in a network. Throughput is essentially similar to digital bandwidth consumption; it may be analyzed mathematically by applying queuing theory, where the load in packets per time unit will be denoted as the arrival rate, and the throughput, in packets per time unit, will be denoted as the departure rate. The throughput of a communication system can be affected by various factors, including the limitations of used analog physical medium, available processing power of the system components, and end-user behavior. While various protocol overheads are taken into account, useful rate of the transferred data made to be significantly lower than the maximum achievable throughput; the useful part can be usually referred to as good put.

Packet delivery Ratio: the ratio of the number of delivered data packets to the destination. This illustrates the level of the delivered data to the destination.
∑ Number of packet receive / ∑ Number of packet send

The ratio of packets that will be successfully delivered to a destination compared to the number of packets, which have been sent out by the end.

Collaborative attack rate: Collaborative filtering is the process of filtering for patterns using techniques including collaboration among multiple agents, viewpoints, data sources, etc. Applications of collaborative filtering typically involve large data sets.

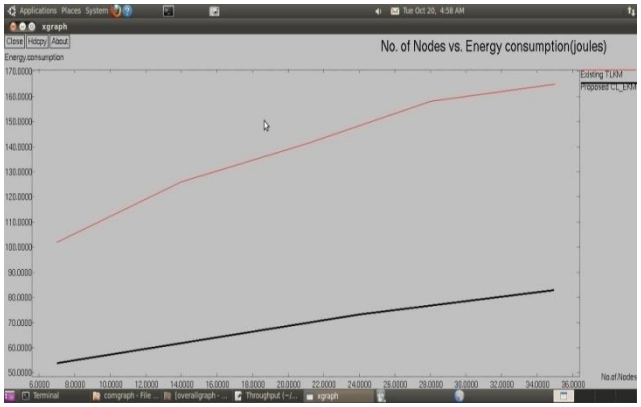

Figure 4: Comparison of existing and proposed systems based on energy consumption.

6. Conclusion

Although the state-of-the-art studies are realized the importance of trust systems efficiency in WSNs and proposed several preliminary solutions, overlooked to EKM technique, which is perhaps among the top secure key managements. The existing, two-layered method of key management scheme and a dynamic key update protocol in dynamic WSNs based on the Diffie-Hellman (DH), respectively. However, since each node must exchange the certificate to establish the pair-wise key and verify each other's certificate before use, the communication and computation overhead increase dramatically. Here it is unable to access with large size of keys and it increases the overhead. Here we cannot provide more security. Resolve the key escrow problem. Here presented a certificate less effective key management (CL-EKM) scheme for dynamic WSNs. In this certificate less public key cryptography (CL-PKC), the user's full private key is a combination of a partial private key generated by using a key generation center (KGC) and the user's own secret value. The special organization of the full private/public key pair will remove the need for certificates and also resolves the key escrow problem by removing the responsibility for the user's complete private key. We also take the advantage of of ECC keys defined on an additive group of a 160-bit length as secure as the RSA keys with 1024-bit length.

7. Future Enhancement

Even though here presented a method of certificate less effective key management, here the BS also suffer from mere problem poor encryption. Since it has four pairs of keys it is not a serious issue. Still the user or the beneficiary authority has to go for more securely encrypted key methods. This problem can be revised and solved and hence to improve this idea of secure data handling. Encryption improvement is the only method to get the most secured way of communication.

References

[1] Ismail Butun, Salvatore D. Morgera, And Ravi Sankar "A Survey Of Intrusion Detection Systems Inwireless Sensor Networks" IEEE Communications Surveys & Tutorials, Vol. 16, No. 1, First Quarter 2014.

[2] Andrea Tassi, Francesco Chiti, Romano Fantacci, and Fabio Schoen " An Energy-Efficient Resource Allocation Scheme for RLNC-Based Heterogeneous Multicast Communications" IEEE Communications Letters, Vol. 18, No. 8, August 2014.
[3] Andrea Tassi, Francesco Chiti, Romano Fantacci, And Fabio Schoen "An Energy-Efficient Resource Allocation Scheme For RLNC-Based Heterogeneous Multicast Communications". IEEE Communications Letters, Vol. 18, No. 8, August 2014.
[4] Bo Zhu, Member, IEEE, Sanjeev Setia, Sushil Jajodia, Senior Member, IEEE, Sankardas Roy, Member, IEEE, and Lingyu Wang, Member, IEEE. "Localized Multicast: Efficient and Distributed Replica Detection in Large-Scale Sensor Networks" IEEE Transactions On Mobile Computing, Vol. 9, No. 7, July 2010.
[5] Taekyoung Kwon, *Member, IEEE,* JongHyup Lee, *Student Member, IEEE,* and JooSeok Song, *Member, IEEE* . "Location-Based Pairwise Key Predistribution for ireless Sensor Networks" IEEE Transactions On Wireless Communications, Vol. 8, No. 11, November 2009.
[6] Wei-Shou Li, *Student Member, IEEE,* Tung-Shih Su, *Student Member, IEEE,* and Wen-Shyong Hsieh. "Multi-Neighbor Random Key Pre-Distribution: A Probabilistic Analysis". IEEE Communications Letters, Vol. 13, No. 5, May 2009
[7] Azzam I. Moustapha, *Member, IEEE,* and Rastko R. Selmic, *Member, IEEE*. "Wireless Sensor Network Modeling Using Modified Recurrent Neural Networks : Application to Fault Detection". IEEE Transactions On Instrumentation And Measurement, Vol. 57, No. 5, May 2008.
[8] Wenliang Du, Member, IEEE, Jing Deng, Member, IEEE, Yunghsiang S. Han, Member, IEEE, and Pramod K. Varshney, Fellow, IEEE."A Key Predistribution Scheme for Sensor Networks Using Deployment Knowledge". IEEE Transactions On Dependable And Secure Computing, Vol. 3, No. 1, January-March 2006

Author Profile

Mr. **Kiran. E** received his B.E degree in Electronics and Communication Engineering from CMS College of Engineering and Technology, Coimbatore, Affiliated to Anna University, Chennai, Tamilnadu. Currently he is pursuing M.E degree in Applied Electronics from Rathinam Technical Campus, Coimbatore, Affiliated to Anna University, Chennai, Tamilnadu. His research interests are networking and wireless communication.

Ms**. T. Nivethitha** received her B.E degree in Electronics and Communication Engineering from Sriram Engineering College Chennai. She received her M.E degree in Applied Electronics from Sri Eshwar College of Engineering and Technology, Coimbatore, affiliated to anna university Chennai. Currently she is serving as the assistant professor in ECE department, Rathinam Technical Campus.