The Comparison of Key Management in Hierarchical and Distributed Wireless Sensor Networks and Providing New Method for Key Management Based on Tree with the Use of Exploratory Algorithm A

Nahid Babaei¹, Mohamed Ali JabraeiJamali²

¹Graduate student of Islamic Azad University
²Computer Department of Islamic Azad University

*Corresponding Author: Nahid Babaei

Abstract

Wireless sensor network are composed of some small physical devices that called sensor, which they duty is collecting information from arrange of study and reporting the results to the requesting center. Although these networks have many advantages for easy installation in hazardous and inaccessible environments, have some disadvantages such as low security about unauthorized access to data, limitations in energy resources, memory and processing speed. For this reason, the need to explore different ways of routing and choosing a suitable method to reduce energy consumption in the networks is necessary. These networks have wide applications and thus are vulnerable about serious security problem. Therefore, to maintain security in this network that is one of the main challenges in wireless sensor networks is considered, we need apply encryption method with the help of create and publish secret key within the network. So that surveying different methods of key management and selecting appropriate solution is one of the essential needs of today. But limited nature of sensors and dynamic performance of this kind of networks prevents the normal use of key management in modern networks. For this reason, many lightweight key management solutions have been proposed to overcome these limitations. In this article we review the state of technical solutions and based on appropriate criteria evaluate two types of them and finally, we will provide a more efficient solution. The most common pre-distributed key management schemes, our focus will be on them. Thus will study, features such as security and network lifetime that can be used to develop some criteria.

Keywords: Wireless sensor networks, Key management, Exploratory algorithm A.

Introduction

Wireless sensor networks, includes a large number of small special devises which they duty is monitoring and sending collected data with the use of wireless channels to the exporter command centers. This small devises called sensor nodes, and have limitation in price, size and resources such as energy, memory, speed and bandwidth, and for gathering information on where the user is not physically present can be used. These networks are independent and autonomous and working without human intervention and working together effectively to meet the general purpose of network. Performance of wireless sensor networks is to report events that occur, to the observer which is not necessary to know the structure of the network and their relationships. These networks are independent and autonomous and working together effectively to meet the general purpose of network (Zunkizank et al., 2010). Wireless sensor networks composed of several sensors and one or two base station.

We should note that all nodes are not capable connected to base station and with help of other nodes can do this, this problem resulted from node limited battery energy. These networks can also appear in larger scale because they are dynamic and nodes can be in motion constantly. For that reasons one of the most important advantages of sensor networks is the ability of manage connections between nodes that are moving. Sensor nodes usually use a battery for power supply which in most cases is not rechargeable and replaceable. Therefor it is critical that to do operation which reduce energy consumption and increase the lifetime of the sensor nodes. There are several ways to access this issue. One of these methods is that to take the number of nodes to sleep mode which is a popular and efficient way. The second method is that to set the transmission range of the nodes corresponding to the neighboring node so that the sensing and transmission of information only be required from neighboring node distance. When
sensor nodes are placed in a hierarchical structure, cluster of nodes can collecting information and, respectively, transmits to the base station.

By this works the responsibilities of routing and data transfer of intermediate nodes is reduced and would have longer lifetime. Efficiency of data collection and routing can also have a significant role in reducing energy consumption. If multiple nodes collecting the same information certainly we will have high energy consumption and wasted. However redundancy elimination is another very important issue in wireless sensor networks (Informatics Society of Iran, 2012). Due to the available hardware, wireless sensor networks can exist in two forms of below: homogeneous and heterogeneous. In homogeneous networks, all of the sensor node in terms of energy, connection power, and other parameters are similar to each other but in heterogeneous networks, usually the more powerful nodes (in terms of energy, radio radius, amount of saved energy, error resilience) are available that calls cluster which can collect and classify the data be obtained by weaker nodes until the desired program process and send the data to the main station. Figure 1 shows samples of homogeneous and heterogeneous nodes in wireless sensor networks (Informatics Society of Iran, 2012).

(a) Heterogeneous (b) Homogeneous

Figure 1. A sample of wireless sensor network.

On the other hand, based on the role that each node has in the network, we can classify wireless sensor networks in to hierarchical and distributed. The aim of the hierarchically network is that each node has a different role depending on its capacity whereas in distributed networks all of the nodes have a similar role. Since the data transfer takes place by the air, to prevent eavesdropping and also prevent damage exposure in these networks we should apply items such as expanding security encryption mechanism in them. Since the initial encryption is done by secret key, we require the use of techniques for secure distribution of keys. Sophisticated encryption solutions that are commonly accepted in modern systems not only require a lot of time to run but also consume a lot of energy. On the other hand wireless sensor networks are very dynamic (that is node within the sensor repeatedly changes). For that reason considering the flexibility is essential for key distribution (Simplique et al., 2010).

In this networks the hardware of sensor types imposes restrictions on the network that we should consider in selecting of routing method such that the power supply are limited in nodes and practically it is not possible to replace or recharge it, therefore, the proposed routing method in this networks should be the best use of available energy that is should be informed of the node resources and if node have not enough resources does not send data to it thereby we able to reduce energy consumption (Wikipedia, 2014). In order to examine the possible performance of these networks they must be simulated. One of the most common method of simulation, is simulation by the use of graphs. In these graphs, each node represents a sensor node and each edge represents a link between two nodes in a sensor network. Figure 2 is a sample of a simulated wireless sensor network with the help of graphs. In wireless sensor networks, traffic basically is a form that data moving from multiple nodes into one node, therefore, the network topology must be done carefully. The first step for topology management is setup of first network that nodes previously had no connection when translocation and starting initial work can communicate with each other.

Figure 2. A sample of a simulated wireless sensor network with the help of graphs.
Topology management algorithms in the initial setup should providing allowing to new nodes join and delete nodes that are disabled for some reason. Topology mobility is one of the properties of the sensor networks that its security challenges. Providing dynamic topology management procedures so that security is also included, is one of the topics in this article that we will review and compare some of them. Since the data transfer takes place by the air, many of these applications require the use of a safety measure to prevent confidential eavesdropping and system destroying by the opponents and adversaries. This can be achieved by extending the encryption mechanism. Since most of the initial encryption need secret keys to running their operations, using techniques which have secure method for distribution of keys is essential (MarkousSimplique & coworkers 2010). Sophisticated multiple purpose encryption solutions that typically accepted in modern systems not only need more running time, but also consume more energy. Moreover wireless sensor networks are very dynamic which means that the position of a node may frequently change. This is due to their relative mobility, battery exhaustion, adding new nodes, etc. For that reasons, observance of principles of resiliency and key distributing is necessary (Simplique et al., 2010).

Comparison of symmetric and asymmetric algorithms encryption

Essentially symmetric key and asymmetric key encryption has two different natures and also has different applications. So comparing these two types of encryption, regardless of the application and the system will not be smarter work. Research show that the length of the message which can be encrypted with symmetric algorithms is less than public key algorithm and symmetric algorithms are more efficient algorithms. But when security issue arise, public key algorithms are more efficient. Shortly we can say that, symmetric algorithms are faster and public key algorithms have better security. The protocols used in the internet, for encryption of keys that are needed to managing public key algorithms are used. If particularly, benchmark comparison, the size and computation time is needed, it must be said with regard to the security scale, symmetric algorithms are very faster than asymmetric algorithms. In general, in simple encryption that the volume of data is enormous symmetric algorithms are used because data encryption and decryption is done with higher speeds. It is noteworthy that, in this paper we have used symmetric key encryption technique in wireless sensor networks (Simplique et al., 2010).

Key management solutions

The research that has been done so far on these networks, key management techniques can be classified in three groups as follows: self-implementation, key agreement schema, pre-distribution schema. Self-implementation is an asymmetric encryption and is used in order to establishment of key after expanding them. Major role of this schema is related to the performance of asymmetric algorithms that currently exist. However, due to low memory and high energy consumption in this way, it is normally not used. Key agreement schema for key to establishment and its managing relies on a trusted central point such as base station. An important issue in this schema is that the central point is easily exposed to risks and in this way the whole network can be disrupted.

When can find trusted point in the network this schema can be considered in terms of security and finally in pre-distributed schema an known entity as the key distribution center, before expansion, will be responsible for keys loading in the sensors that it can be done by physical connections or wireless. The schema can be linked to the networks with little or no dependence on the central station after expansion. That's why this plan is most suitable for wireless sensor networks. That's why this schema is most suitable for wireless sensor networks (Simplique et al., 2010). Since the mechanisms of key management in wireless sensor networks contains a wide range of solutions, that's why we do not intend to cover all international protocols. In this article we examines Hierarchical key management and perspectives of pre–distribution and based on the criteria and applications that are they have will be assessed them.

Hierarchical wireless sensor networks

In these networks, each node, depending on its ability obtain one of the following roles: base station, head cluster and sensor node. Sensor node is limited availability in the network that lonely is responsible for gathering data from the surrounding environment and send them to the nearest cluster heads. Head cluster usually have more resources than sensor nodes and their main purpose is to collect and combine past read issues from nearby sensors and as well as is routing the resulting data to the base station. The base station collects and processes the incoming data and then sends the results to other networks. Most of these stations are able to reach all nodes in the network that's why they are much more powerful than other entities in hierarchical sensor networks, such as sensor nodes and cluster heads (Simplique et al., 2010).
For establishment of hierarchical sensor network, sensor nodes after deployment are grouped into clusters and each cluster contains one cluster head and a number of sensors. One of the main challenges in wireless sensor networks, is bandwidth and energy consumption. With the using a hierarchical architecture we can increase network scalability and thereby lead to reduced communication overhead, delay time and also the complexity of managing. Hierarchical routing protocols in the wireless sensor networks are well known. One the protocols is LEACH routing protocol that classify wireless sensor networks into several clusters, and sensor nodes within a similar cluster and can be caused direct contact with each other. Clusters based on the received signals decide to join which cluster. After connecting to clusters the sensor nodes gathering information and send it to base stations within the same cluster, are randomly select the cluster heads. Since cluster heads consume more energy to reduce power consumption are constantly being replaced (Yang et al., 2010).

**Distribution and key management in hierarchical wireless sensor networks**

Using traditional security mechanisms to prevent attacks on wireless sensor networks with the given limited resources of a sensor node is difficult. In recent years, many security mechanisms have been proposed, that one of them is encryption method to provide authentication and key management. In the next state we can note key distribution and revocation. To establish a secure communication between sensor nodes, key distribution is done and key revocation carried out at the time of attacks to the network and compromising the keys. Currently, mostly of the key pre-distribution is used to keys management in wireless sensor network. As shown in Figure 4, the simplest method of key distribution, are use of same key for all the sensor nodes. In this method, sensor nodes are pre-loaded by the master key so that both nodes can use it for encrypted communication. In this case, if the network is vulnerable to attack, attacker easily with holding of the master key can control the whole network (Yang et al., 2010).

**Figure 3. A sample of hierarchical wireless sensor networks**

Another radical approach of above schema is that, it is necessary for both nodes share a unique pair key. This is shown in Figure 5. In this way when network is compromise all nodes in the network do not compromise and it does not compromise the entire network. In a network with n node we should have (n-1) key and the entire network will store n (n-1)/2 key. In this case, if the n is too large number this will needed much more memory to store the pair key in the network (Yang et al., 2010).

**Figure 4. A key advantage of the network**

In random key pre-distribution for creating a key ring, for a node randomly selects k keys from the key pool and each key contains an ID number. After nodes deployment to find nodes share the same key, Each node must expand key ID in its communication range. According to figure 6 so as long as there is at least one key shared a secure connection will be established. If there is no common key between two nodes, connective link will be created through two or more key route (Changlung Yang and coworkers 2010).
In comparison with the shared pair key, this method requires fewer keys before loading and can thus save memory space. This method according to multiple links and with the shared key cannot authenticate the existence of neighboring nodes. It also not requires a lot of computation time to create a pair key but for searching shared key time will be proportional to the number of keys (Yang et al., 2010).

Initialization phase

Before the deployment each node should be inter to the initialization phase. In this form base station functions is as a trusted certificate authority. The base station for each node creates a public and private pair key and to ensure the identity of added nodes, issued relevant certificate. The base station that is shown in table 1, system parameters initialized with selecting a large number of P, as well as a and b parameters of the elliptic curve initialized to define the group of elliptic Ep(a,b). Then the main point G of the elliptic group is choosing with a very large amount. Base station established its own public and private keys in accordance with $K_{PUB(BS)} = K_{PRI(BS)} \* G$ and also for every node established public and private keys in accordance with $K_{PUB(ID)} = K_{PRI(ID)} \* G$. Then ID number and public key $K_{PUB(ID)}$ will be store for future use. Every node pre-loaded by ID number, public key $K_{PUB(ID)}$, private key $K_{PRI(ID)}$, the main point $G$, elliptic group Ep(a,b), public key of base station $K_{PUB(BS)}$ and also certification. In Figure 7, overview of certificates to confirm the initialization is presented (Yang et al., 2010).

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ID_i$</td>
<td>identity number of node i</td>
</tr>
<tr>
<td>$BS$</td>
<td>base station</td>
</tr>
<tr>
<td>$F_p$</td>
<td>finite field, p is a large prime number (&gt;160bits)</td>
</tr>
<tr>
<td>$Ep(a,b)$</td>
<td>elliptic curve equation</td>
</tr>
<tr>
<td>$G$</td>
<td>a base point on the elliptic curve</td>
</tr>
<tr>
<td>$n$</td>
<td>order of base point $G$</td>
</tr>
<tr>
<td>$K_{PUB(ID)}$</td>
<td>public key of node i</td>
</tr>
<tr>
<td>$K_{PRI(ID)}$</td>
<td>private key of node i</td>
</tr>
<tr>
<td>$K_{PUB(BS)}$</td>
<td>public key of the BS</td>
</tr>
<tr>
<td>$K_{PRI(BS)}$</td>
<td>private key of the BS</td>
</tr>
<tr>
<td>$C_i$</td>
<td>certificate of node i</td>
</tr>
<tr>
<td>$T_i$</td>
<td>time stamp</td>
</tr>
<tr>
<td>$M$</td>
<td>Message</td>
</tr>
<tr>
<td>$r$</td>
<td>a random number</td>
</tr>
<tr>
<td>$MAC(K,M)$</td>
<td>message authentication code</td>
</tr>
</tbody>
</table>

After deployment, head cluster can be considered as new certified, and the base station of head cluster public key $K_{PUB(ID)}$ will be used for encryption. So in this case, the private key of head cluster $K_{PRI(ID)}$ can be used to decrypt and obtain a certificate. When
one head cluster obtain a new certificate, will become it as a center of trusted certificates of a cluster and so the other node in the cluster can act accordingly to issue a new certificate. In the figure 8, after deployment become a cluster heads and also become a new certificate of base station. In accordance with public key of node A, the base station for encryption after confirming its authenticity, will be issued new certificate. Thus, only nodes that have private key of node A are able to decode the new certificate (Yang et al., 2010).

![Figure 8](image)

Figure 8. Confirmed cluster heads with the new certificate.

According to figure 9 when a cluster head is a trusted certificate center, other nodes of this cluster can use it to apply a new certificate. Cluster heads confirms certificate and use public key of node \( K_{PUB(ID)} \) for encryption and issue a new certificate. For example, the node B apply as a certificate of cluster head A, such that will confirm certificate and then will use public key of node B for encryption until be sure to get a certificate node B can decrypt it (Yang et al., 2010).

![Figure 9](image)

Figure 9. Applied to the sensor node with the new certificate.

According to figure 10 selected new head cluster of B, will be sent its current certificate to the base station for approval. The base station also for encryption will be issued new certificates with help of public key of head cluster \( K_{PUB(ID)} \) for encryption and issue a new certificate (Yang et al., 2010).

![Figure 10](image)

Figure 10. Applied cluster heads with the new certificate.

After selecting head cluster other sensor nodes should again issue a new certificate through the head cluster. According to figure 11 node A will apply new certificate from the head cluster (node B). Node B for issuing new certificate is used the resulted public key, then from base station will use for checking the validity of certificates of node A. Finally public key of node A will be used for encryption to ensure that node A can decrypt the obtained certificate (Yang et al., 2010).

![Figure 11](image)

Figure 11. Applied to the sensor node with the new certificate.
Distributed wireless sensor networks

In these networks for every node independently nothing has been issued instead the sensors are similar and the relationship between two pairs of nodes can occur in the network. Moreover we will review the criteria for evaluation of management techniques in these networks. It should be noted that some of these criteria are very common and some of them are requirements imposed on the system. In any case, these criteria can be classified into three main groups: Security, efficiency, flexibility (Simplique et al., 2010).

Figure 12. An example of a distributed wireless sensor networks.

Security

As regards in the networks private keys should be provide by a safe method so that activity of malicious groups on the network is neutral, so we should ensuring that only trusted entities are able to update keys in the network and are prevented external resources from doing this work. Moreover ways should be predicted to prevent the unauthorized disclosure of keys that a number of them are: Node authentication, resilience, node elimination. In other word this technique should guarantees that nodes in a safe method are able to identify each other. This feature helps network to detect node that has error function (authentication). It also should points to key management schema against the node capture, for example when the nodes are physically attacked by the enemy and want to recover data from its memory is interrupted by this criterion that was met a network attack except communication where compromised nodes are directly involved in, that in this case, the network will continue to operate (resilience). Finally discovered the nodes in the network which are compromised and providing management solutions to remove them from the network. The mechanism is used for preventing the insertion of malicious nodes in the network (node elimination) (Simplique et al., 2010).

Efficiency

Each network must be capable to establish required keys for the proper conduct of operations on the network and at the same time consider limitations for them. In this regard, key establishment should not import abundant load about memory, bandwidth, processing, energy and key links to the network. Namely the amount of memory needed to store data in the network as well as the number of processing cycles for keys isn't high and moreover isn't need high-bandwidth for data exchange in networks also energy consumed by the nodes should not be too high. When a group of sensors are able to create a shared key, connection will be possible in two form, first it only take place between two pairs of adjacent nodes (localized connection) and second connection occurs in the general case of the network, in this case inappropriate key links seriously compromise network performance because the adjacent nodes can't be connected securely (Simpliqu et al., 2010).

Flexibility

In many applications, sensors are spread dynamically and randomly that this function makes it difficult to identify the position of the nodes and even after spreading the node location information is collected, so key generation techniques in terms of flexibility is not related to the position of the nodes. On the other hand in the lifetime of the network, it is possible sensor dynamically change so the key distribution schemes should have support the large networks and the introduce new nodes without loss of security (scalability). Because in key establishment schema one basic common key is loaded on all sensors, this creates efficient and flexible network and so regardless of the network size very little memory to store keys will be needed (Simplique et al., 2010).

Key management based on a tree structure

As regards in the asymmetric encryption different keys are used for encryption and decryption operations some systems allow the public key is published but the private key is kept by the its owner and however the sender is aware of the content of the main information but it can’t achieve the original source from coded source and therefore coded message for every receiver with the
exception itself receiver would be meaningless. While in symmetric encryption method one key is used for encryption and decryption so the length of the message which is encrypted by symmetric algorithms will be less. It also has more processing speed and in computing that the volume of data is enormous will perform better because data are speedily encrypted and decrypted. Our proposed key management method based on tree structure is one of the symmetric encryption methods that it can be have all the above mentioned advantages and so resolve many challenges that can be addressed in the context of wireless sensor networks (Muoungkim et al., 2008). Since secure key exchange in wireless sensor networks is one of the issues that in addition to security problems, in terms of energy and load processing on the sensors has always been considered. Due to the limited processing power and the amount of energy in those nodes, another component of study in key exchange, is energy consumption of this exchange process. In this article tree based management with the use of exploratory algorithms A* are presented.

**Exploratory algorithms A**

In computer sciences, algorithms A* is a computer algorithm that widely be used in graph traversal and finding a path between two points that is called sensor. Due to the high performance and accuracy of this algorithm, it is widely used. This algorithm in fact is a generalization of the Dayjsra algorithm and with the using of exploratory operation the resulting outperformed in terms of time. This function helps to decide which node is the best case for the development of next node. Indeed with help searching first best, finds shortest path between source and destination node is given by the other node. This method by 1 equation evaluates the minimum cost path between the nodes $n_0$ to the target node:

$$f(n) = g(n) + h(n)(1)$$

In which, $h(n)$ function representing minimum cost of path between the node $n_0$ to the target node and also $g(n)$ is minimum cost of path value from $n_0$ ton. Thus we see that, minimum cost of path value $n_0$ to the target node is more than all paths leading to the node $n$. $f(n)$ function also representing the estimation of cheapest solutions by n cost. For any node exploratory agent $h^*(n)$ estimate some of $h(n)$ and depth agent $g^*(n)$ is amount of resulted minimum path cost by A* ton node. In other words, if we're looking for the cheapest solution, the first reasonable job is the check of node which has the lowest amount $f(n)$. It is clear that this strategy is quite reasonable. If $h^*(n)$ be in the certain conditions then searching A* will be complete and optimized. If A* be used with tree searching algorithm, then it becomes easier to analyze the optimality of it. If $h^*(n)$ function is an acceptable exploratory function, A* will be optimized, that is $h(n)$ function never estimated the cost of achieving the target more than actual amount. This exploratory function is optimized inherently because they think that the value of the solution costs less than the actual amount. Since $g(n)$ accurately show the cost of achieving n, we can immediately conclude that $f(n)$ never do not overestimate the value of true cost solution that passes through n (Muoungkim et al., 2008).

$$f^*(n) = g^*(n) + h^*(n)(2)$$

The time complexity of the algorithm A* depends on its arguments. At worst, the number of nodes in the search space of the target level, towards solving is exponentially. If the error of expletory functions grows faster than the actual costs of the logarithm, exponential growth occurs. In other words, condition of exponentially small grows is as follows:

$$|h(x) - h^*(x)| = O(\log h^*(x))(3)$$

In which $h^*(x)$ is the achieving real cost from node n to the target node (Muoungkim et al., 2008). In this method the number of re-key messages reduced and in the result will increased network stability and longevity of the network. This method has been proposed to deal with safe polymorphism and is causing efficiently update of the group key. In this way, also with considering the history of deleted or added nodes and the branching factor at each level that can be used on different values, we reduced the number of re-key messages. The number of re-key messages when adding and removing nodes in a network, have a direct relationship with tree height and by reducing the height of the key tree also the number of re-key messages are reducing (Muoungkim et al., 2008).

The key management method which is presented in this paper due to the topology of the it’s structure, in the case of network attacks can be easily removed nodes that have been compromised from the network without compromising the whole network and when we need to expand the network scale again, we can easily add new node to the network without we concerned that key information may be at risk and so network security is provided (Muoungkim et al., 2008).

The method of tree structure based on exploratory algorithm A*, have been investigated security mechanism for key management and credit providing for identity transmission of provided message and performance of computation time and power consumption. In the wireless sensor networks before deployment, by issuing certification such as newly added nodes which are not required to update their keys, sensor nodes are pre-loaded. As long as the certificate and stored keys size are fixed in the nodes, number of storage nodes in the network will have no effect (Muoungkim et al., 2008).

**Discussion**

In this study, we have been investigated security mechanism for key management and credit providing for identity transmission of provided message and performance of computation time and power consumption. In the wireless sensor networks before
deployment, by issuing certification such as newly added nodes which are not required to update their keys, sensor nodes are pre-loaded. As long as the certificate and stored keys size are fixed in the nodes, number of storage nodes in the network will have no effect. Because both hierarchical and distributed method have many problems and for the reasons that listed above, are not proposed for key management of wireless sensor networks so we have covered another way namely tree structure which is one of the symmetric key methods which partly solve the problems have mentioned above.

This study tries to evaluate the nodes authentication, transmission of provided message, performance of computation time and power consumption in the wireless sensor network and thus cause a group key management in wireless sensor networks. Because the keys are created and published as a tree structure we can by the help of data history and also branching factor raised, thereby minimizing the tree height reduces the number of re-key messages and ultimately increase the lifetime of sensor networks and so one of the major problems in wireless sensor networks are solved. Thus with reducing the amount of tree height we can easily remove and add new nodes in the network to establish the number of re-key messages and reduce the number of them and by this work the life of the network increases. Because A* algorithm holds all nodes which is created in memory usually rather than having low time, having low memory so computation time is not the main disadvantage of the this algorithm. Moreover because A* explatory algorithm in many of the big issues isn’t scientific, for that reason we should be considered strategies to solve this problem of key management in wireless sensor networks and as well as with the adoption of new strategies solved the problem of low memory.

References

http://fa.wikipedia.org/wiki