Sensors & Transducers, Vol. 186, Issue 3, March 2015, pp. 33-42

Sensors & Transducers

http://www.sensorsportal.com

A Novel Situation Specific Network Security for Wireless Sensor Networks

[1] Mohammad Al-Rousan, [2] Muneer Bani Yassein, [3] Ahmed Al-Dubai, [2] Barraq Ghaleb, [2] Ibrahim Mahmoud

[1] College of Computer Science and Engineering, Kuwait University, Kuwait
[2] Faculty of Computer & Information Technology, Jordan University of Science and Technology, Irbid, Jordan
[3] School of Computing, Edinburgh Napier University, Edinburgh EH10 5DT, UK
E-mail: [1] Alrousan@just.edu.jo, [2] Masadeh@just.edu.jo, [2] ibrahim_mufeed@just.edu.jo, [2] barraq_ghaleb@just.edu.jo, [3] A.Al-Dubai@napier.ac.uk

Received: 22 December 2014 /Accepted: 27 February 2015 /Published: 31 March 2015

Abstract: Researchers have come up with many high level security protocols and key management algorithms that can be applied for different types of networks. The main characteristics of wireless sensor networks that are limited computation and power made it difficult to apply these secure key management algorithms since it consumes much power and resources. Sensors in the field at different periods of time of their life time need higher level of security more than other times - sensors in a battlefield need higher security when it is war time. In our research we developed an adaptive security protocol for two-layer clustered heterogeneous wireless sensor networks that can toggle between five different modes of operation, each with different security level. Each level of security has its own encryption/decryption algorithm and specified key length depending on the situation of the application and the situation for the wireless sensor networks. *Copyright © 2015 IFSA Publishing, S. L.*

Keywords: Sensor network, Security, Key management, Clustered heterogeneous.

1. Introduction

Wireless sensor networks consist of too many sensor nodes that are distributed in a field and have physical capabilities to measure or sense things in the real world, do some computations, communicate with each other and deliver intended results to a base station. These sensors have limited computation and a limited power as well. Network life time depends on its sensors life time, and the sensor life time depend on its power consumption [18, 20].

The wireless sensor network has many unique characteristics that differentiate it from other types of wireless network; those unique characteristics include the following:

- Mobility of nodes;
- Dynamic network topology;
- Communication failures;
- Heterogeneity of nodes;
- Large scale of deployment;
- Ability to cope with node failures.

One of the most challenges facing this type of networks is that the sensors are very small in size and consequently have less power than other electronic devices like computers and laptops. So we should be careful about the communication between the sensor

nodes such that they can consume less power and forward the messages to the destination in the least possible cost [16-17, 19].

Wireless sensors may be deployed in hostile areas, thus they will be a target for different types of attacks; an adversary can easily eavesdrop on wireless communication to gain confidential information, also an adversary not only can eavesdrop the traffic in a network, but also can intercept or interrupt the exchanged messages. Thus achieving secure communication paradigm in sensor networks become an important issue.

For achieving secure communication in wireless sensor networks secret keys should be used to encrypt wireless communication and establish data confidentiality integrity and authentication among sensor nodes. Therefore, it is necessary to develop appropriate security mechanism for wireless sensor network to distribute secret keys into sensors, encrypt wireless communication and establish authentication among sensor nodes. However, the challenge does not relies on the development of secure mechanisms only, but also on how to efficiently generate, distribute and maintain secret keys among sensor nodes specially that we know that; distributing keys inefficiently and implementing an inefficient key management schema may consume the constrained energy in wireless sensors and as a consequence minimizing the lifetime of the WSNs. Thus an efficient key distribution and management schema is required to maximize the lifetime of WSNs.

Different key management and distribution protocols have been designed [1-2, 11], to secure the communications between the sensor nodes themselves and between the sensor nodes and the base station. All of these protocols try to trade-off between the security and the constrained power of the WSN sensor nodes, but what we noticed about these protocols that they are assuming the same security requirement (level) throughout the whole lifetime of the wireless sensor networks. This may be not true for many applications and environments where the level of security may change as a function of time and place. The security level for these protocols varies depending on the encryption algorithms used and the length of the used keys. During our investigation in this field we did not find any research that considers changes in the security level for the WSNs.

In this research we developed an adaptive security protocol for two-layer clustered heterogeneous wireless sensor networks that can toggle between five different modes of security levels, each level of security has its own encryption/decryption algorithm and specified key length depending on the situation of the application for the wireless sensor networks.

In our new protocol we have three main contributions; the first contribution is developing a method for constructing the clusters in heterogeneous wireless sensor nodes. The second contribution is developing a key distribution schema. While our third contribution, which is our main contribution in this research, is developing an adaptive key management schema by allowing the WSN to use different key lengths and different encryption/decryption algorithms according to the required level of the security. In reality our third contribution can be applied or run over any key distribution schema.

In this scheme we have used three types of network elements: the base station, the cluster head which we called it H_sensor node, and the ordinary sensor which we called it L_sensor node. We assumed that each H_sensor node has much power in term of computations, memory and power resources than the ordinary L_sensor node. And they can communicate in between to establish pair-wise key using asymmetric key exchange algorithm. The H_sensor node is responsible for aggregation, compression, adaptive redistribution of the new keys to the ordinary sensor nodes in cooperation with the base station and sending the collected data to the base station. The L_sensor is limited in term of computation, memory and power resources.

The rest of the research is organized as follows. In the next section, we identify the problem statement. In Section 3 we review some related work in wireless sensor network security. In Section 4, we describe our new protocol which we called it The Message. The security, energy consumption and storage aspects of this protocol are analyzed in Section 5. Finally, in Section 6, we conclude this research and discuss some future work.

2. Problem Statement

As we said in the previous sections that security problem in wireless sensor network has been attracted many researchers, because of the fact that sensors are deployed randomly in unprotected areas and as a consequence they will be subject to different types of attacks, those attacks can be divided into passive and active modes. In passive mode, the adversary can eavesdrop wireless communications among sensor nodes to capture mission critical or private information. In active mode, adversary may capture some sensor nodes in the networks, when some nodes are compromised then the keys of the networks will be revealed and the adversary now can control the whole network in some way. So, it become necessary to develop a strong security protocol for distributing and managing of the encryption, decryption and authentication keys in the network.

So, applying very secure algorithms for WSN consumes much of the limited sensors energy and effects the lifetime of the network negatively. On the other hand applying other types of security protocols that consume less power from sensor nodes makes it easier on attackers to attack the network. Finding a way that best secure the WSN while maintaining the

power consumption of the sensor nodes is the dilemma that we are trying in this research to figure out an accepted solution for.

3. Related Work

During our literature review we found that there are many security schemes designed for the wireless sensor networks [1-2, 11]. The first security scheme is a simple one that assumes wide network key [3]. All the nodes in the network share this key in order to establish a secure communication link.

Simplicity of this approach make it good choice when we are seeking high network performance and/or long lifetime. However, on the other hand once the wide key is known, the security of the entire network is dropped. Some other protocols suggest to dynamically change the key, but this solution consumes the network bandwidth and also power of the sensors as it require extensive communication.

Another network scheme is trusted base station [4] that suggest any two nodes need to communicate securely have to request the base station for a session key. The network bandwidth is not utilized in this approach because large numbers of the messages are key-request messages. At the mean time the base station will be busy answering nodes key-requests and this will increase packet delay.

Other researchers proposed a pool of keys [5] that contains a number of keys. Each sensor node at the deployment of the network is given a number of these keys to use for its secure communication. A key discovery phase is done so that each node knows which neighbour nodes share the same keys with. Any two nodes that share the same key from the pool can communicate securely between each other. If a node need to communicate with another node that does not share the same key with, a secure bath must be established between them first, where any two nodes on that bath share a common key.

Many deterministic key management protocols have been suggested [7-8], the most famous one is the Localized Encryption and Authentication Protocol (LEAP) protocol which have been proposed by Zhu, *et al.* [7]. LEAP includes support for multiple keying mechanisms:

• LEAP supports the establishment of four types of keys for each sensor node;

• An individual key shared with the base station;

• A pair-wise key shared with another sensor node;

• A cluster key shared with multiple neighbouring nodes;

• And a group key shared by all the nodes in the network.

Most key management schemes were designed for homogeneous WSNs. However, such networks have poor performance and scalability, many schemes designed for homogeneous sensor networks suffer from high computation and communication overheads.

At this time, there are only few key management schemes for heterogeneous WSNs, For instance, Boujelben, *et al.* [1] proposed a key management scheme for heterogeneous two-tiered WSNs. Their network model consists of two tiers, the upper tier contains sensors with high capabilities (H-sensors) and in the lower tier contains the sensors with low capabilities.

They proposed to use a different key management protocol at each tier. In the lower tier they used the random key distribution method combined with Blom scheme [8]. In the upper tier, they applied the pairing Identity based cryptography (IBC) which is an ECC type public key cryptographic to distribute pair-wise keys.

There are also many key management protocols have been designed for clustered wireless sensor networks [1, 6, 9-11]. For instance, In Sec LEACH [10], Leonardo B. Oliveira, *et al.* Proposed to add security into the LEACH - like protocols. Their approach was based on probabilistic key distribution method. The scheme was used to establish key pair between the nodes and its cluster head but they do not take care about the keys between the cluster heads and the base station. They proposed to generate a large pool of S keys and their ids in advance to network deployment. Each node is then assigned a ring of m keys drawn from the pool pseudo-randomly then after deployment they can communicate to setup and distribute node-to-cluster-head keys.

Another key management for the cluster WSNs is the Logical Ring Based which is proposed by A. S. Poornima and B. B. Amberker [6]. In their paper they have proposed two schemes for key management in clustered sensor networks with changing cluster head. The Simple Secure Logical Ring (SSLR) scheme, they proved that SSLR communication and computation cost needed for key establishment is constant.

They also proposed another schema called it (BDLR) scheme key establishment which is based public key cryptographic mainly the Elliptic curve [12], they indicated that BDLR schema could be achieved by performing (n+1) multiplications and (n+5) communications, where n is the number of nodes in the cluster.

Also Reza Azarderakhsh, *et al.* [11] have proposed a key management schema for Cluster Based Wireless Sensor Networks. In their protocol they used public key cryptography, as in the previous schema mainly they used the Elliptic curve cryptographic to establish a shared key between sensor nodes and gateways. Each node requests a session key from the base station to establish a shared key with its neighbours after clustering phase.

Also Zhang, *et al.* [9] proposed A Cluster-Based Group Key Management schema, their method allow only to the cluster heads to generate and distribute the group key (the group key here refers to a scenario that sensor in a group can send and receive messages

from group members) to the sensors within the cluster. They constructed a bivariate polynomial for each group over a finite field and use it to construct the keys by exploiting the symmetric property of this type of polynomials.

4. Detailed Overview of Our Schema

In our research, we proposed a novel adaptive security protocol, in heterogeneous clustered sensor networks; the new protocol will provide different security levels according to the situation of the WSN, which the base station specify.

4.1. Wireless Sensor Network Architecture Model

The wireless sensor network that we assumed here is depicted in Fig. 1; we assumed a two-hop clustered heterogeneous wireless sensor network consists of two types of sensor nodes as well as the base station.

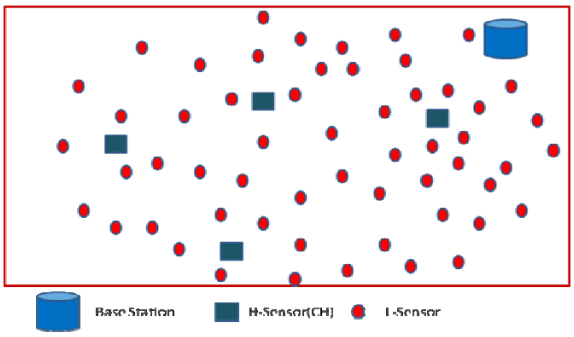

Fig. 1. Wireless sensor network deployment model.

The base station (BS): which is the ultimate destination of all the data generated by all sensor nodes, it is also responsible for controlling the security level of the network depending on the information coming from the sensors or on explicit request from the observer.

The H_sensors: these sensors have powerful resources in term of the storage and the power, so they are responsible for aggregation and compression the data coming from L-sensors then sending the aggregated data to the base station, they are also responsible for doing extensive computation operations. They are also responsible for distributing the key materials to other sensor nodes within the wireless sensor network.

The L_sensors: these sensors are constrained in term of the storage and the power and the general function of each L-sensor node is to collect the data from the surrounding environment then sending the collected data to the closest H_sensor which is the cluster head.

4.2. Creating the Clusters in our Model

Before we can go into creation of the cluster steps we should first say that the number of Sensor node will be pre-determined, in our model we assumed the number of the H_sensors will be 5 % of the total number of the sensor nodes deployed in the sensing field, another thing we should say that the cluster heads could be only chosen from the H_sensor nodes.

We suggested also in our model that the number of the clusters will be equivalent to the number of H_sensor nodes, for more understanding see the following example:

Example: If we deployed 100 sensor nodes then the number of H_sensor nodes will be 5 nodes (for 200 then number of H-sensors will be 10) which is equal to the number of the clusters that will be created after deployment.

After the deployment of the nodes in the sensing field the following steps will be conducted to create the clusters:

1. Every H_sensor node will consider itself Cluster Head (CH) and as a consequence it will send a broadcast ADV message to all of the sensors in the field, this message will contains its ID.
2. Every L_sensor nodes that received the messages from all the cluster heads should choose one of H_sensors to be its cluster heads, depending on the least communication cost, then sending Join message to the chosen cluster head.
3. Now each CH will create schedule for its members and send them this schedule which contains the time slot for every member, this node will send the data during the time slot specified for it.
4. Now every L-sensor during the time slot specified to it in Step 3 will send the collected data to its cluster head, then the cluster heads send the data to the base station.

4.3. Assumptions

Our sensor network model is based on the following assumptions:

1. We assumed that all the sensors are not mobile, this mean they cannot change their positions after deployment.
2. The base station will never be a compromised, it is equipped with tamper resistance materials.
3. The H_sensor nodes have more storage and power than L_sensor nodes which have a limited power and memory, thus they can use public key cryptographic to setup and distributes keys between of every pair of them.
4. Each L_sensor nodes can only communicate with the H_sensor nodes and never with an L_sensor node.
5. Every sensor node is provided with an identification (id) number and the Sensors are sensing the environment at a fixed rate and thus

always have data to send to the CHs and then to the BS.

4.4. The Message Protocol

In this section will show how The Message protocol works, our protocol mainly consists of three parts, the first part is how to create the clusters after deployment and how to choose the cluster heads which is our first contribution. In the second part is the concern about how to distribute the keys for the sensors nodes which is our second contribution in this research, the third part is concerned about how to manage those keys i.e. update them, when a change in the level of security has been occurred -this is our main contribution in this research. Fig. 2 shows the high level description of our protocol.

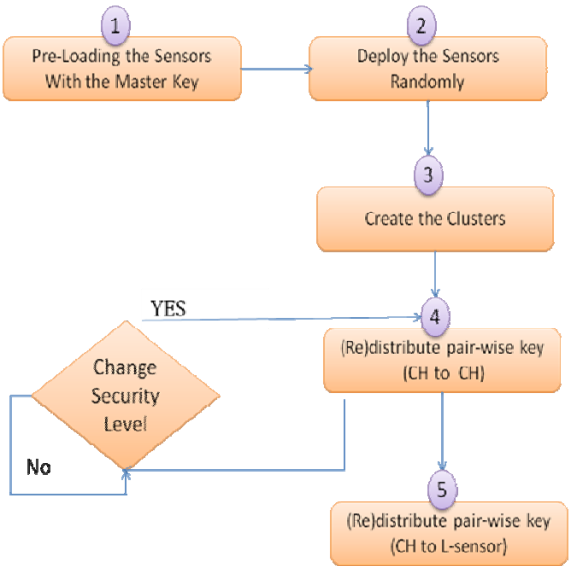

Fig. 2. The high level description of the message.

We should say here that our main contribution schema -the third one, is ultimately independent from our clusters creation schema -the first contribution, and key distribution schema -the second contribution, therefore it can be combined with any key distribution schema.

4.5. Clusters Creation

We described in previous subsection how to create the clusters in our model of wireless sensor networks, so what we need to add here that: Before deployment all the sensors will be pre-loaded with the same master key to be used during the steps of clusters creation for encryption, decryption and authentication of the exchanged messages between the nodes.

4.6. Initial Key Distribution

As we said in the previous section our network consists of two layers, the first layer where the communication is conducted between the cluster heads (CH to CH), and (CH to BS), because all the sensors in this layer have a high energy resources, then they can use the new Elliptic Curve Diffie-Hellman Two-party Key Agreement Protocol (EECKE - 1) developed by Eun-Jun Yooni and Kee-Young Yoo in 2010 [12] to exchange a pair-wise key between Every pair of the CHs and also between each CH and the BS, but we restrict using of this distribution scheme only when high security level is required and there is no a safe way to exchange the keys other than this, else we can use the same mechanism used in the second layer to update the keys (or let one of the CHs or the base station distribute new keys).

The following steps show how we can use this public key algorithm to setup a pair-wise key between a two H_sensor nodes N and M.

1. N generates its private key k_N and computes its public key $K_N=k_N\times P$, where P is a chosen point on the used elliptic curve, then N sends K_N to node M. Analogously M generates its private key k_M and computes its private key $K_M=k_M\times P$, then M sends K_M to node N.

2. N picks a random number $R_N\leq n$, where n is the prime order of the point P on the chosen curve. And computes $Q_N=R_N\times K_M$. Analogously, M picks a random $R_M\leq n$ and computes $Q_M=R_M\times K_N$.

3. $N\rightarrow M:Q_N$, this meant N send Q_N to M.

4. $M\rightarrow N:Q_M$.

5. *N* and *M* compute, respectively, the points T_N and T_M as follows:

$T_N=h\times k_N\times R_N\times Q_M$

$T_M=h\times k_M\times R_M\times Q_N$,

where h is the cofactor of the elliptic curve, see ref(). Those distinct computations will produce point $T_N=T_M$.

6. Finally N and M will compute the shared pair-wise key SK by using a key derivation function kdf as:

$SK=kdf(T_N,Q_N,Q_M,ID_N,ID_M)$ at node N,

$SK=kdf(T_M,Q_N,Q_M,ID_N,ID_M)$ at node M.

To establish a pair-wise key between (L-sensor, CH), before deployment each sensor has the master key and they can communicate using the Blondo schema [14] as follow:

To setup a pair-wise key between sensor I and its cluster head H, they exchange their node id's first encrypted and authenticated using the master key, then node I calculate f(I,y) at y=H, and the same calculation is applied by the cluster head H to calculate its polynomial share f(H,y) at y=I, because of the symmetry property of the 2-biavaruate polynomial, both the sensor I and its cluster head H will get the same value (f(I,H) – f(H,y)), which can be used as their pair-wise communication key between the sensor I and its cluster heads H.

4.7. Situation Specific Key Management

Having the sensors deployed in the sensing field and they used the Initial Key Distribution schema as described in the previous section to establish pair-wise keys in the network, then it is the time to explain how we can update and redistribute new keys as a result of the change in security level.

1. Once a change in the security level has been accrued, then [Change-Security] message should be sent by the base station to all cluster heads or to a subset of those cluster heads, where there is a change in security level, informing them the required level of security.

2. Depending on the required level of security, each cluster head who received Change-Security message should act by creating new keys of the required length and change the type of encryption algorithm used, this will happen according to Table 1.

3. Now each cluster head will distribute the new keys to its member.

4. Each cluster head which received the [Change-Security] message will run the EECKE-1 protocol to re-setup a pair-wise key compatible with required level of security in that cluster with all other cluster heads in the network. If both cluster heads are already using a key with a higher security level, then this step will be skipped. Also each cluster head which received the [Change-Security] message will generate pair-wise keys between its members and the other cluster heads, those keys will be used in case of cluster head capturing.

Table 1. List of used simulation parameters.

Security level	The Key Length in bits	The Used algorithm
1	256	AES
2	128	AES
3	64	RC5
4	32	RC5
5	-	NO Security

Now we can give a detailed scenario to better understand our key management and distribution protocol which combine our three contributions as follow see Fig. 2 for a high level description of our protocol:

1. Before deployment of the sensors in the deployment field, all sensors will be preloaded with a master key (MK), which will be used after deployment to setup a pair-wise key between every H_sensor and L_sensor and also to setup and create the clusters after deployment.

2. Also each H_sensor node will be preloaded with a pair-wise key to communicate with the base station, in this stage, type of the encryption algorithm and the key to be preloaded is depending on the security level at that time, (i.e. if the security level is the second level at that time then we will pre-load the nodes with MK of the length 40 bits and to use the RC5 as encryption or decryption algorithm). Table 1 shows the security levels and the corresponding keys associated with each level.

3. After deployment, the clusters must be created as explained in previous section. We just add here that all the exchanged messages during clusters creation will be encrypted, decrypted and authenticated using the preloaded master key (MK).

4. After cluster setup and creation, each H_sensor node should setup a pair-wise key with each L_sensor node, here will come the role of key distribution that we suggested in previous sections, also we can setup a pair-wise key between each two H_sensor nodes ,by using EECKE-1 key exchange Protocol. After key distribution all nodes should delete the master key from their memories and use the pair-wise key shared between them for communication.

Note: We assumed that the security level will not change during Step 3 and Step 4, and also we assumed the time needed to Steps 3 and 4 is T1 and the time needed to compromise any node is T2, we assumed that T1<T2, so the adversaries during those steps cannot compromise the master key (MK).

5. Now the network will continue working in normal situation and no update will occur until one of the following condition happen:

a. An explicit request from the base station to any cluster head informing him to change the level of security in the specified cluster.

b. The effect of adversary attack goes beyond a specified threshold which is determined in advance.

This second condition will be cluster-dependent decision, this mean that the cluster head will cooperate with its members to detect compromised nodes, and we assumed here that we assumed here that; within the cluster and according to those information, the cluster head will decide how to adjust the level of security within its group.

6. Once one of the conditions stated above has been accrued, then the network should react to the change in the security level by change the type of encryption/decryption algorithm and the key length, and therefore redistribution of new keys is required and will go as described in previous sections.

5. The Message Protocol Analysis

The most important security issue in the wireless sensor network is compromising the nodes (capturing the node physically and then reading the content of its memory to get the pair-wise keys shared with other nodes). Hence, our focus in this section will be on security analysis of our protocol against this real problem.

5.1. Security Analysis

- **Node Compromised**

In our protocol after deployment, each cluster head will setup a pair-wise key with each L-sensor node and also after updating the keys. Thus if an adversary capture one node it will get only the shared keys between this node and each one of the CHs, but it cannot compromise the other keys for the nodes within the same cluster or in the other clusters and because we assumed that cluster head is able to detect the compromised using the same mechanism in [15], the cluster head now should delete the pair-wise key shared with the compromised node, also the CH of the compromised node should send the ID of this node to all cluster heads so that they can also delete the shared pair-wise key with the compromised node, this will mainly prevent the attacker from deploying the same node once again in the network.

- **Cluster Head Compromised**

Also our protocol is secure against capturing the cluster head physically, when detecting that the cluster head is captured then the node which discover this should send alert message to the BS telling it that its cluster head has been captured, now the BS send a message to closest cluster head, this message will contains the id of the compromised cluster head, asking it to communicate with the nodes whose CH has been captured – this each CH should store the ID of every node and its cluster head.

5.2. Storage Analysis

In our protocol the overhead of storage keys is minimized in comparison with other sachems, first each L_sensor node should only store pair-wise key with each H_sensor (CH) node as well as a pair-wise key with the base station(this key will be used only in case of the L-sensor detect a compromised node). So, if we assumed that the number of nodes in the network is N, then the number of the Cluster Heads will be $0.05 \times N = HN$, then the number of pair-wise keys every L-sensor will store is NPK which will be as follows:

$$NPK=0,05 \times N + 1 = HN + 1$$

This is a compatible with the constrained memory of the L_sensor nodes, so the total number of keys in L_sensors (LN) that will be in the network omitting the pair-wise key with base station will be $((LN) \times HN)$.

On the other hand, because that each H_sensor node has high storage, then they can store a pair-wise key with each node in the network, so if we assumed that the number of the nodes in the network is N, then the number of keys H_sensor will store is N and the total number of keys that will be stored in all the H_sensors will be $((HN - 1) \times N)/2$. Then the total number of keys will be as follow:

Total Num = Number of keys in L_sensors + number of keys in H_sensors

$$(((LN) \times HN) + ((HN-1) \times (N)))/2$$

5.3. Energy Consumption Analysis

Because our protocol is adaptive one, then energy consumption will be minimized because of the following reasons:

- Changing the length of the key and the type of algorithm.
- Our choosing to the cluster heads is static, and this mean that there is no communication cost is needed to re-elect the cluster heads.

6. Simulation Result and Analysis

Security solutions for sensor networks involve trade-off between security level and resources consumption. In this Section, we evaluate energy consumption of our proposed schema. We simulated network of different populations. Table 2 shows used simulation parameters.

We have run our simulator with many different configurations, and the results showed that our new approach -we called it hybrid, produce good enough results that make the WSN lifetime longer, keeping high security level. We compared our schema with efficient pair-wise key establishment and management scheme in [13] which is the random pair-wise schema.

Table 2. List of used simulation parameters.

Deployment Region	100 × 100 m
Data packet Size	2000 bit
Security Change	Variable (1, 2, 10,100, 1000, 2000) second
Simulation Time	Variable (1000, 2000, 5000) second
Number of Nodes	Variable (100, 200) second
Schedule Time	1 second

6.1. The Message Energy Analysis

Before demonstrating the energy consumption analysis we may give the following concepts:

- Low: refers to the lowest level of security, this mean that we run the simulator with no security requirement throughout the whole life of the network on our key distribution management.
- Moderate: refers to the third level of security, this mean that we run the simulator with third level of security throughout the whole life of the network on our key distribution management.
- High: refers to the highest level of security, this mean that we run the simulator with highest level

of security throughout the whole life of the network on our key distribution management.

- Pair-wise: means that we run the simulator with random pair-wise key management, because this protocol has no adaptive mechanism, we assumed that it always use the high level of security.
- The numbers appearing in the Time bar in the figures represent the time unit in minutes.
- The numbers appearing in the Energy bar in the figures represent the energy unit in Joule.
- Hybrid: refers to our protocol simulation, this mean that we run the simulator with changing security level from 1 to 5 as in Table 1 on our key distribution management.
- Pair-wise: means that we run the simulator with random pair-wise key management, because this protocol has no adaptive mechanism, we assumed that it always use the high level of security.
- The numbers appearing in the Time bar in the figures represent the time unit in minutes.
- The numbers appearing in the Energy bar in the figures represent the energy unit in Joule.

In Fig. 3 We have Run the simulator on 100 sensor nodes, for 2000 seconds time, and the security level changed every 10 seconds, From that figure the result was as expected, the Low mode has the lowest energy consumption, while the random pair-wise mode has the highest energy consumption, the figure show also that our protocol (Hybrid) work better than the random pair-wise key management in term of energy consumption. The same results were obtained from Fig. 4, and Fig. 5 However, with different parameters.

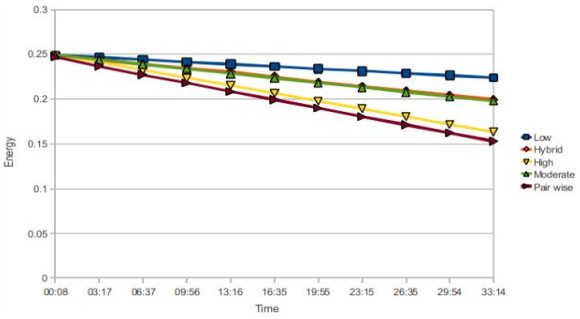

Fig. 3. 100 nodes, 10 intervals, 2000 time.

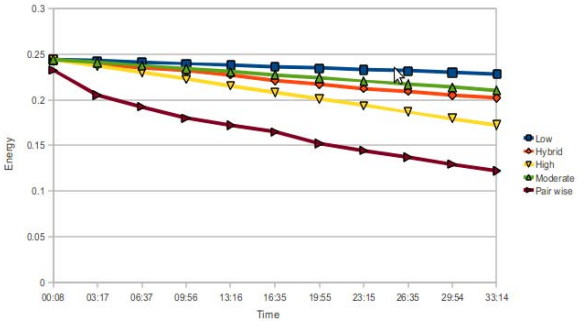

Fig. 4. 1000 nodes, 2 intervals, 2000 time.

Running the simulator on 1000 sensor nodes, for 2000 seconds time, and the security level changed every 2 seconds. Results in Fig. 4.

Running the simulator on 1000 sensor nodes, for 2000 seconds time, and the security level changed every 10 seconds. Results in Fig. 5.

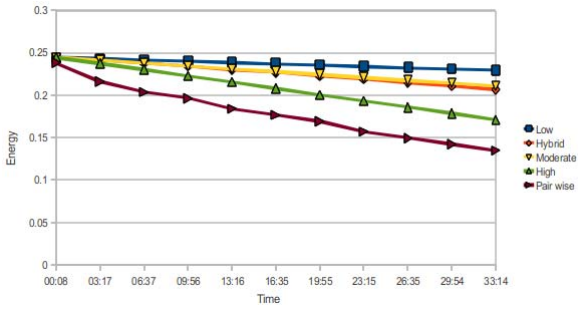

Fig. 5. 1000 nodes, 10 intervals, 2000 time.

In Fig. 6 we have run the simulator with different security interval time(the time the base station should wait then change randomly the security level). From that figure it appears that when we increase the interval time (with keeping the other parameters constants) the energy consumption will be minimized, this result is understandable and logical, that is because when we increase the interval time this mean that we minimize number of communication to update the keys and as a consequence the energy consumption.

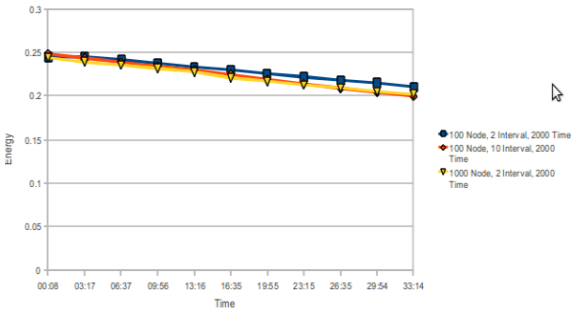

Fig. 6. Comparing the hybrid with different interval.

We have run the simulator with longer simulation time for 12000 second (200 minutes) to figure out the first and last died node in our protocol in comparisons with pair-wise protocol. The results are presented in Fig. 7 (the time unit are). These results show that our protocol has increased the life time of the network by order of magnitude. Comparing our protocol with different security changing interval namely (2 seconds and 200 seconds) appears in Fig. 8.

We run the simulator also with longer simulation time (12000 seconds = 200 minutes) the result of our protocol in comparison with Pair-wise presented in Fig. 9.

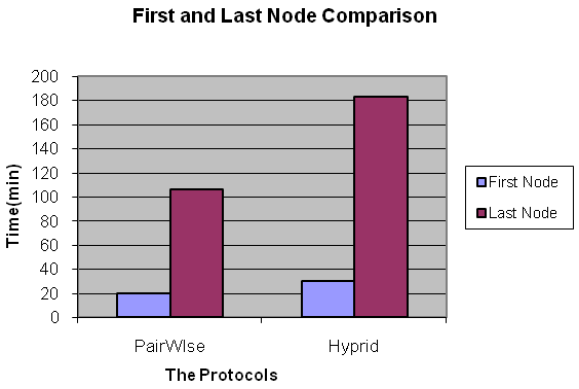

Fig. 7. Comparing network life time.

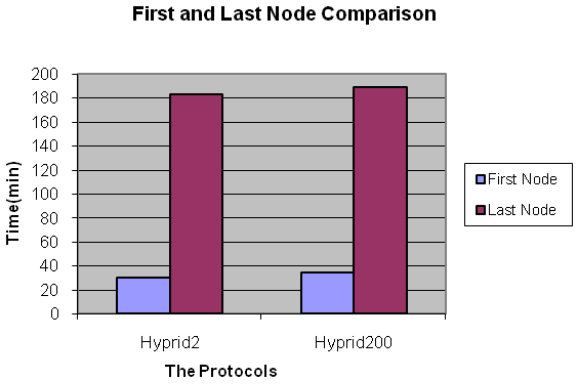

Fig. 8. Last and first died node.

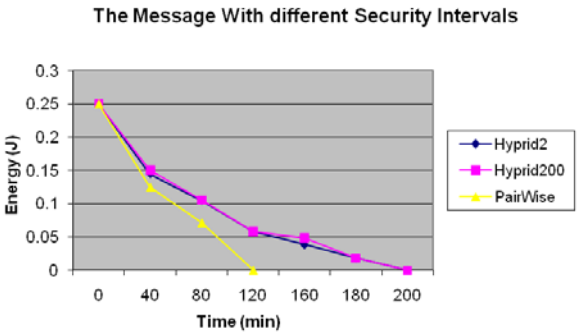

Fig. 9. Pair-wise and our protocol with longer time.

7. Conclusions

In this research we have designed a new security protocol for key management and distribution in heterogeneous clustered wireless sensor network, our protocol consists of three phases, the first phase is how to create the clusters, the second phase is how to initially distribute the keys and the third phase is how to manage and redistribute the keys during the life time of the network. Securities, Storage, Performance analysis are performed to demonstrate the feasibility of our solution for wireless sensor networks.

In the near future we are planning to investigate our protocol in homogeneous wireless sensor network where all the nodes have the same constrained resource, we are planning to adjust mainly the cluster creation phase, our first contribution in this research, and key distribution phase, our second contribution in this research, while keeping the key management phase as it, our third contribution in this research.

References

[1]. Boujelben M., Cheikhrouhou O., Abid M., Youssef H., Establishing pairwise keys in heterogeneous two-tiered wireless sensor networks, in *Proceedings of the IEEE 3rd International Conference on Sensor Technologies and Applications*, 2009, pp. 442-448.

[2]. Kee-Bum K., Yi-Ying Z., Wen-Cheng Y., Myong-Soon P., An Authentication Protocol for Hierarchy-Based Wireless Sensor Networks, in *Proceedings of the 23rd ACM International Symposium on Computer and Information Sciences*, Istanbul, Turkey, 2008, pp. 1-6.

[3]. Chan H., Perrig A., Song D., Key distribution techniques for sensor networks, *Kluwer Academic Publishers*, Norwell, MA, USA, 2004.

[4]. Xiao Y., Rayi V., Sun B., Du X., Hu F., Gallowa M., A survey of key management schemes in wireless sensor networks, *Computer Communications*, 30, 2007, pp. 2314-2341.

[5]. W. Mardini, M. B. Yassein, Y. Khamayseh, B. A. Ghaleb, Rotated Hybrid, Energy-Efficient and Distributed (R-HEED) Clustering Protocol in WSN, *WSEAS Transactions on Communications*, 13, 2014, pp. 275-290.

[6]. Poornima A., Amberker B., Logical Ring based Key Management for Clustered Sensor Networks with Changing Cluster Head, in *Proceedings of the International Conference on Signal Processing and Communications (SPCOM'10)*, Bangalore, India, 2010, pp. 1-5.

[7]. Zhu S., Setia S., Jajodia S., Efficient security mechanisms for large-scale distributed sensor networks, in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, Washington, DC, USA, 2003, pp. 62-72.

[8]. Blom R., An optimal class of symmetric key generation systems, *Advances in Cryptology: Proceedings of EUROCRYPT 84, Lecture Notes in Computer Science*, Springer-Verlag, France, 1984, pp. 335–338.

[9]. Zhang Y., Shen Y., Lee S., A Cluster-Based Group Key Management Scheme for Wireless Sensor Networks, in *Proceedings of the 12th International Asia-Pacific Web Conference (APWEB'10)*, Busan, 2010, pp. 386-388.

[10]. Oliveira L., Wong H., Bern M., Dahab R., Loureiro A., SecLEACH – A Random Key Distribution Solution for Securing Clustered Sensor Networks, in *Proceedings of the 5th IEEE International Symposium on Network Computing and Applications*, Washington, DC, USA, 24 July 2006, pp. 145-154.

[11]. Azarderakhsh R., Reyhani-Masoleh A., Abid Z., A Key Management Scheme for Cluster Based Wireless Sensor Networks, in *Proceedings of the IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, Shanghai, 17-20 December 2008, pp. 222–227.

[12]. Yooni E., A New Elliptic Curve Diffie-Hellman Two-party Key Agreement Protocol, in *Proceedings of the 7th IEEE Society International Conference on Service Systems and Service Management (ICSSSM'10)*, 28-30 June 2010, Tokyo, pp. 1-4.
[13]. Cheng Y., Agrawal D., Efficient pairwise key establishment and management in static wireless sensor networks, in *Proceedings of the 2nd IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, Washington, DC, USA, November 2005.
[14]. O. Younis, S. Fahmy, HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, *IEEE Transactions on Mobile Computing*, 3, 2004, pp. 366-379.
[15]. Marti S., Giuli T., Lai K., Baker M., Mitigating Routing Misbehavior in Mobile Ad Hoc Networks, in *Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, New York, USA, 2000, pp. 255-265.
[16]. R. Luchmun, M. Pyanee, K. K. Khedo, Hierarchical Hybrid Energy Efficient Distributed Clustering Algorithm, *International Journal of Computers and Distributed Systems*, 2, 2012, pp. 10-20.
[17]. D. Wei, Y. Jin, S. Vural, K. Moessner, R. Tafazolli, An energy-efficient clustering solution for wireless sensor networks, *IEEE Transactions on Wireless Communications*, 10, 2011, pp. 3973-3983.
[18]. S. Sendra, J. Lloret, M. García, J. F. Toledo, Power Saving and Energy Optimization Techniques for Wireless Sensor Networks, *Journal of Communications*, 6, 2011, pp. 439-459.
[19]. J. N. Al-Karaki, A. E. Kamal, Routing techniques in wireless sensor networks: a survey, *IEEE Wireless Communications*, 11, 2004, pp. 6-28.
[20]. S. Sendra, J. Lloret, M. García, J. F. Toledo, Power Saving and Energy Optimization Techniques for Wireless Sensor Networks, *Journal of Communications*, 6, 2011, pp. 439-459.
[21]. T. Scientist, Title of the paper, in *Proceedings of the Conference on Smart Sensors and Systems: Technology and Applications (SSS&TA'03)*, Paris, France, 16-19 March 2003, pp. 123-127.