I.J. Information Technology and Computer Science, **2012, 12, 50-59**
Published Online November 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2012.12.05

Architecture Aware Key Management Scheme for Wireless Sensor Networks

Benamar KADRI[1], Mohammed FEHAM[1], Abdellah MHAMMED[2]

[1/]STIC Lab., Department of Telecommunications, University of Tlemcen, Tlemcen, Algeria
[2/]Telecom Sud Paris, France
E-mail: benamarkadri@yahoo.fr

***Abstract*—** The emergence of wireless networking as well as the development in embedded systems and technologies have given birth to application specific networks called wireless sensor networks WSNs, their flexibility, facility of use and deployment as well as their low cost give them an increasing field of applications. Usually sensors are limited in capacities deployed in a hostile and unpredictable environment, making the security of these networks a challenging task. In this paper we are going to present a key management scheme in which the base station play the role of the secure third party responsible of distributing key and managing security in the network, two versions of this scheme are presented the first one for flat networks and the second one for hierarchical networks in which the cluster head play the key role in all key agreement with the base station.

***Index Terms*—** WSN, Clustering, PKI, Security, Cryptography, Key Agreement

I. Introduction

A wireless sensor network is composed of hundreds to thousands of small, low cost, low power and multifunctional sensor nodes, having the possibility to sense and collect application-specific data like temperature, pressure and movement to allow environment monitoring [1, 2].

Due to their facility and flexibility of deployment, ad hoc connectivity as well as the autonomy and the cost-less of sensors, wireless sensor networks are implied in several fields ranging from the military applications such as battlefield surveillance [3], to civilian ones including environment and habitat monitoring, healthcare applications, home automation, traffic control, environmental monitoring, or to detect and characterize Chemical, Biological, Radiological and Nuclear in some environments where the presence of human is not possible[4].

Wireless sensor networks are very often part of unpredictable and hostile environment, usually exposed to many risks and attacks. Therefore, the aspect of security must take the greatest attention to protect the network from the increasing number of attacks against WSNs, under the constrained nature of sensors, usually limited in energy and computing power which makes the conventional security schemes useless for WSNs.

In literature several strategies were proposed to secure WSNs, however the majority of them are based on symmetric encryption due to resources consumption of asymmetric encryption which made them vulnerable against several attacks. However, the recent advance of sensor technologies as well as the development of new energy efficient encrypting algorithm make possible to partially use asymmetric encryption to guaranty additional security services such as authentication and integrity which are not possible using the symmetric encryption alone.

In this paper we present architecture aware key management scheme based on asymmetric encryption in which the base station plays the trust authority over the network, responsible of distributing keys. Two versions of the key management scheme are proposed, the first one is intended to be executed over flat networks and the second one is intended to hierarchical networks.

II. Sensor Network Architectures

In a wireless sensor network, sensors are very often dispersed in a large region usually without any centralized authority for managing the network architecture and establish connectivity from end sensors to the base station; therefore sensors must collaborate between themselves to establish this connectivity without the help of any administrative authority. Classically, two main architectures exist for WSNs the hierarchical and the flat network architectures [6].

2.1 Flat Network Architecture

In flat network architecture, all nodes are equal in roles and connections are setup directly between nodes and the base station, in the way that data is sent from sensors to the base station such as any other ad hoc

network. Sensors use traditional routing to establish end to end connection with the base station.

Flat networks are very suitable for stable sensor network where the collected reports are not numerous, which do not add a great overhead to the network for their transmission to the base station. However, routing in such architecture uses flooding which consumes lot of network resources and occasionally overhead the network.

2.2 Hierarchical Network Architecture

In a hierarchical architecture, sensors are organized into clusters or regions. One node in the cluster is elected as cluster head intended to manage the cluster formation and maintenance, other sensors called cluster members are attached to the nearest cluster head.

Cluster heads manage the transmission between sensors and the base station which minimize considerably the network overhead since the collected data is sent to the cluster head which sent an abstract report to the base station which minimizes the traffic out clusters and therefore over the network.

III. Security in Wireless Sensor Networks

Security is a very important issue when designing or deploying any network or protocol, however the recent developed networks such as the wireless ones have not given the necessary attention to security when designing protocols by taking into account the specificity of these networks such as the used medium and the devices constraints as well as the environment of deployment [7].

In this section, we are going to discuss the various key management schemes provided in literature as well as their feasibility and effectiveness for WSNs.

***Shared Key*:** This is the simplest scheme to secure any class of network wired or wireless [8]. In this kind of schemes one key is shared between all the network nodes. In the case of WSN the shared key is preloaded by an offline dealer before the network deployment. From resources point of view this scheme does not add any overhead for key exchange and establishment since the used key is stored in each sensor before deployment also very little computation and storage capacity are needed to encrypt and decrypt traffic over the network. In the other hands this schemes is very vulnerable against cryptanalytic attacks, since a WSN is deployed for long period which makes this class of attacks possible. In addition to cryptanalytic attacks, this scheme present a point of failure which is the shared key since the compromising of a single node due to physical attack causes the compromise of the whole network.

Pair-wise Key Establishment: In order to overcome the shortcoming of the previous scheme caused by the use of a single shared key, the pair-wise key establishment propose to share different keys between each pair of sensor over the network, the used keys are preloaded before deployment by an offline administrative authority. Therefore, for a network of n sensors, every node stores n−1 keys in its memory. After deployment each node establishes a secret key with each one of its neighbors, by looking for the corresponding key in the pull of keys preloaded before deployment [8].

This scheme does not need any communication and computation resources since the keys are preloaded before deployment, however it impose a large storage capacities in order to store all the possible keys, making this solution unsuitable for large scale networks.

From the security point of view this scheme ensures a great threshold of security robust against several attacks.

Random Pair-wise Key Establishment: This scheme is developed to overcome the shortcomings of simple pair-wise key establishment, which needs an important capacity of storage to store all the possible keys. This scheme assumes that all pairs of sensor nodes in a WSN do not need a communication path with each other, in the way that a node need to secure only a sub set of links [10].

To make this in practice sensors store only a sub set of the key pull defined in simple pair-wise key establishment and shares secret key with its neighbors with a given probability. This probability must be chosen according to the number of sensors in the network as well as the desired level of connectivity, which makes this scheme more complex and does not manage efficiently the network widening. Compared to the simple Pair-wise Key Establishment this scheme is more efficient.

Trusted Key Distribution Center: In the previous key management schemes each sensors shares a secret key with each of its neighbors, in order to minimize the overhead due to communication however these schemes are not suitable for large networks, due to the limited storage capacity of sensors [11]. Accordingly, the trusted key distribution center mechanism proposes to minimize the overhead due to key storage by installing a central server responsible of key distribution over the network. After the network deployment each sensor contacts the server to obtain a pair-wise key for every session. This scheme is robust against node capture and traffic analysis since the key may be updated periodically. However, it adds a high communication overhead for pair-wise keys establishment which causes an area of congestion around the key server. It also creates a point of failure which is the trusted server against spoofing attacks, since the spoofing or cloning of the server may compromise the whole network.

LEAP: Localized Encryption and Authentication Protocol was proposed to manage the confidentiality and the authenticity of the network using four keys, each one with a special purpose:

- The first one is shared with the base station and used for securing communication with the base station.
- A broadcast group key shared with all the network sensors and used for group communication such as broadcast messages.
- Pair wise key shared with other sensor nodes,
- A cluster key shared with a number of sensors in a given neighbourhood.

This scheme make a compromise between the overhead due to storing a great number of keys such as in pair-wise key establishment and the overhead due to communication such as in trusted key distribution center presented above. By storing a predefined keys used to derive and establish other keys to secure links between each pair of sensors. However, this scheme still vulnerable against capture attacks, since the compromising of one node makes the network subject of lot of attacks such as flooding attacks since the broadcast key is shared with all the network nodes.

Secure Pebblenets: This solution [13] is an extended version of the shared key solution, by proposing to share a set of key rather than a single key. Secure Pebblenets provides group authentication, confidentiality and message integrity, by using symmetric encryption.

For simplification, a hierarchical architecture is used to divide the whole network into clusters, each cluster is managed alone using several encrypting keys for securing intra and inter cluster communication as well as data integrity and authentication.

This mechanism ensures the integrity and the confidentiality of the network using only symmetric encryption which makes it suitable for WSNs, since this kind of cryptography is less resource consumption, however it stills vulnerable against capture attacks as well as spoofing and flooding attacks. In the other hands the scalability of the network is managed efficiently by creating new clusters.

Tinysec: is a link layer security protocol based on symmetric key encryption, TinySec [14] supports two different security options: authenticated encryption (TinySec-AE) and authentication only (TinySec-Auth). TinySec uses link layer encryption to secure hop to hop links which is very efficient against denial of service attack; however this scheme is not energy efficient because the operations of encryption and decryption are done by each sensor over the path between end sensors and the base station. In the other hands, TinySec needs another key management scheme to deliver encrypting keys, which means that this protocol can be used by any other key management scheme as an underlying tool for encryption.

TinyPK: Although, the security schemes developed for WSNs are based on symmetric encryption, because the asymmetric encryption consumes more system resources compared to the symmetric encryption. However, the recent technology progress has given the possibility to use asymmetric cryptosystems such as ECC and RSA to ensure other security services such as authentication.

The TinyPK [15] is designed specifically to allow authentication and key agreement between resource constrained sensors. The protocol is designed to be used in conjunction with other symmetric encryption based protocols such as TinySec, in order to deliver secret key to the underlying protocol using Diffie-Hellman key exchange algorithm. The protocol is designed to use pair-wise key using Diffie-Hellman which consumes a great amount of system resources, especially in large WSNs. However it ensures a good threshold of security with an acceptable resistance against several attacks.

IV. Design Constraints for WSNs

4.1 Network Constraints

Due to the nature of sensors having reduced computing, radio and battery resources as well as the ad hoc paradigm of wireless sensor networks relying on multi hop to ensure connectivity over the network without any infrastructure or centralized authority any protocol should take into consideration the following characteristics of a WSN [16, 17, and 18]:

***Constrained Devices*:** Due to their size sensors are extremely limited in resources (battery power, computing power, storage capacities) which makes the development of applications and protocols for WSNs a challenging task. Therefore the developed protocols and services for WSNs must take these constraints during development by developing efficient and robust security or routing protocols by minimizing the number of operations needed for executing any task.

***Wireless Medium*:** Wireless sensor networks uses radio waves as transmission medium, inherently vulnerable due to its broadcast nature giving possibility to any attacker with the adequate hardware and the network stack to intercept, eavesdrop or modify the exchanged data.

***The nature of environment*:** Generally, a wireless sensor networks are intended for remote controlling and surveillance, deployed in unpredictable and hostile environment, making them subject of many attacks such as sensors capture, compromise and spoofing attacks.

***High number of sensors*:** Future wireless sensor networks will be composed of hundreds to thousands of sensors geographically dispersed in a large area, with limited resources. Therefore any developed protocol must allow the network scaling.

***Absence of infrastructure*:** Although a wireless sensor network is composed of sensor nodes wirelessly linked to each other, responsible of establishing, maintaining and securing the connectivity with the base station without any administrative authority, thus sensors collaborate in a distributed fashion to manage the network security and connectivity.

***Mobile topology*:** Due to the nature of sensors which can be attached to mobile objects as well as the possibility of failure of sensors during the network lifetime, topology changing is very frequent, therefore the stability and the connectivity of the network must be guaranteed under all possible configurations.

4.2 Attacks against WSN

Due to the nature of implied devices as well as the used medium which is the radio waves naturally opened in a large hostile and unpredictable area, wireless sensor networks are exposed to several attacks more than any other networks:

Eavesdropping: this passive attack is the simplest attack against an opened network, in which an attacker with the adequate hardware and software passively listen the exchanged data over the network in order to get information about the structure of the network and the underlying routing protocols which can be used for future active attacks [19].

***Data modification*:** This attack tries to intercept the transmitted data sent from sensors to the base station and modify the final report sent to the base station which can corrupt the whole goals of the deployed network, by sending false reports to the base station [19].

***Sink hole*:** also called black hole attack, its objective is to attract all the traffic from a particular area or node through a compromised node [20], by injecting false routing information advertising the attacker as the legitimate sink or base station which forces the network traffic to pass over the attacker, in order to stop the network service or to execute other attacks such as man in the middle, data modification or eavesdropping, …etc.

***Selective forwarding*:** In selective forwarding attack, malicious nodes may refuse to forward certain messages and simply drop them, ensuring that they are not propagated any more. In contrary of sink hole attack which can be easily detected, in selective forwarding the adversary selectively forwards packets and drop or modifies other packets originating from a defined area or nodes and forwards the remaining traffic which can complicate its detection [20].

***Spoofing attacks*:** also called impersonate attack is executed in the absence of an authentication mechanism, this attack is executed by one or more attackers by spoofing the identity of legitimate sensors in order to gain access to the network using the spoofed identity and execute other attacks. This attack presents a great risk if the spoofed identity is the base station which means that all the collected reports are sent to the false base station [21].

***Denial of service attacks*:** this kind of attacks tries to disrupt, deny, degrade the service of the network, it is planned in different manner and decreases network lifetime in different ways. Among all the denial of service attacks the flooding attack is executed against wireless sensor networks, in which an attacker broadcast permanently hello messages which are rebroadcasted by each sensor over the network, which consumes the network bandwidth and sensor nodes resources which decreases the network lifetime [22].

Sybil Attacks: Similar to conventional network, the attacker presents multiple identities to the rest of sensors by installing new malicious sensors or by using spoofed identities of legitimate nodes. This attack is used to collude a given sensor or area in order to execute other attacks such as black hole or selective forwarding. This attack can be executed against authentication less networks, since a mechanism of authentication can stop or limit the effect of this attack [23].

***Node capture*:** This is a physical attack in which the adversary gains access to the hardware and software of one or more sensors over the network [24]. After the capture of the sensor, the attacker gets all the cryptographic keying and algorithms, which gives him the possibility to listen, interrupt, alter or modify messages. This kind of attacks is usually executed against large sensor networks deployed in a large area or a hostile environment.

False Node: In this kind of attacks, one or more malicious sensors are injected in the network as legitimate nodes. The malicious node tries to act as a legitimate node and occasionally injects false routing data in the network in order to perturb the valid network functioning or simply execute flooding attacks which degrade the network lifetime and performance [24].

V. Public Key Cryptography for WSN

Public key cryptography also called asymmetric cryptography uses two keys for encryption and decryption. In the way that any message encrypted with

one of the keys can only be decrypted with the other key. One of the keys is called private key which is kept secretly by its holder, and the second one is publicly known by each entity in a given community, using these two keys, the public key cryptography can ensure both confidentiality, integrity and authentication.

However, public key cryptography is omitted from the use in WSNs, due to its great consumption of energy and bandwidth which are very crucial in sensor networks.

However, last years have known the development of new cryptographic algorithms more energy efficient and giving the same threshold of security as the conventional algorithms such as RSA. Elliptic Curve Cryptography (ECC) [25] is one of these new algorithms and it is the most promising regarding the energy and time consumption, which makes it very attractive for data encryption in WSNs. ECC offers the same security with smaller key size which saves memory, computational and energy power of sensors.

Table 1: Energy cost of digital signature (mJ)

Algorithm	Sign
RSA-1024	304
ECC-160	22,82
RSA-2048	2302,7
ECC-224	61,54

In the other hands, the new developed sensors will be more powerful concerning the CPU and memory capacities, making public key encryption possible for small sensors in WSN.

VI. Network Architecture and Assumptions

In the remainder of this paper we are going to present a security scheme for wireless sensor networks based on public key cryptography as a tool for managing mutual authentication between sensors and the base station.

Public key cryptography is used in our proposed scheme to guaranty the authentication of the base station since only the base station has a pair of asymmetric keys (private, public), the public key is preloaded for each sensor over the network before deployment, this key is used by sensors to authenticate the base station and secure the handshake, which guaranties the integrity and the confidentiality of all dialogues with the base station, since only the base station has the valid private key for decryption.

In order to manage efficiently the security over the network and to be used for all configurations of the network, we propose to use two versions of our proposed scheme the first one for flat networks and the second one for hierarchical networks.

In order to implement this security scheme we assume that:

- The base station have more computational and energy power compared to sensors.
- The base station has a pair of keys (private and public key).
- Each sensor is capable to use:
- Asymmetric Cryptography: To provide authentication of the base station.
- Symmetric Cryptography: To ensure the confidentiality of traffic across the network.
- MAC (message authentication code) to ensure data integrity.
- Each sensor has the capacity to save at least the public key of the base station and one or more symmetric keys used for data encryption.
- Each sensor receives the public key of the base station by an off-line dealer.

VII. Securing Flat Network

In flat network sensor have the same roles and capabilities, each sensor gets environmental measure and sends it to the base station using the underlying routing protocol. Therefore each sensor is responsible of the connectivity with the base station. Consequently, for our proposed scheme, each sensor launches a handshake with the base station in order to establish a symmetric session key used for data encryption.

7.1 Sensors to Base Station Handshake

In order to secure data communication between sensors and the base station, we propose to establish a secure tunnel between them using a symmetric shared key. This key is established using a handshake encrypted with the public key of the base station to protect it from eventual attacks.

The sensor to base station handshake is executed in two steps:

Handshake launching: Each sensor over the network initiates this operation by generating a random symmetric encrypting key. The generated key is encrypted using the public key of the base station and sent in a regular packet to the base station using the underlying routing protocol.

The use of public key encryption for the handshake guaranties:

- The authentication of the base station, since only the base station has the corresponding private key and can decrypts the message containing the

symmetric key.

- Integrity and confidentiality: no intermediate node can read or decrypt the message containing the symmetric key.

Session key Establishment: After the reception of the message containing the session key, the base station decrypts this message using the corresponding private key.

The base station stores all the keys received from each sensor over the network in a global table, this table is used to identify the sensors and their session keys.

In order to validate the received session key, the base station sends a challenge message for the corresponding sensor. If the corresponding sensor decrypts the challenge message sent by the base station, the handshake is successfully achieved and the two entities can use this key for future communication, otherwise a man in the middle attack is assumed over this route which launches a new handshake using an alternative route.

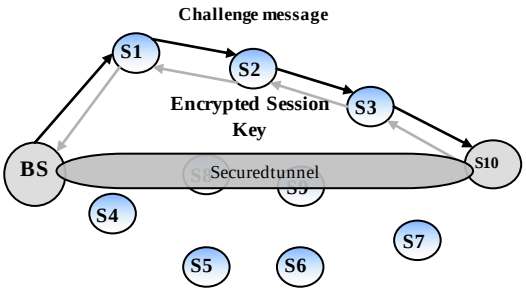

Fig. 1: Secure tunnel over flat network

7.2 System Functioning

After a successful handshake, each sensor shares a secure tunnel with the base station; the tunnel provides both confidentiality and authentication of communications with the base station.

To ensure data integrity, another additional mechanism is used; which is a MAC (message authentication code) joined to each message exchanged with the base station.

Consequently, each packet is passed in a hash function to obtain a fingerprint which is encrypted using the session key shared with the base station. The encrypted MAC is joined to the original packet without any modification on the global structure of the packet.

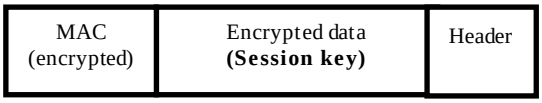

Fig. 2: Packet structure

Another option can be turned on to ensure more security depending on the importance and the nature of the networks is to encryption the MAC joined to each packet using the public key of the base station, this option consumes more energy due to the additional overhead for data encryption however it guaranties a maximum of security.

7.3 Key Update

By nature a wireless sensor network is deployed in a hostile environment or in some area which are not accessible by human, therefore the need for an efficient security scheme which can resist against long term attacks is primordial, so a key update must be executed periodically to enforce security over the network and avoid long term attack aiming to extract the encrypting keys by analyzing the encrypted traffic over the network for long period. The period of the key update is defined according to the complexity of the encrypting algorithm and the length of the encrypting key.

The key update is achieved by launching a new handshake between the corresponding sensor and the base station, which consists on the generation of a new encrypting key used as session key for the next period.

VIII. Securing Hierarchical Network

As we have defined above a hierarchical network is organized into clusters where one of the members of a given cluster is intended to play the role of the cluster head responsible of some management task such as data aggregation, routing or security. Using the same idea presented above for securing sensor network based, we propose to execute a handshake between the bases station and the cluster heads over the network, which will be the base of securing the rest of cluster members.

8.1 Cluster Head to Base Station Handshake

Using the underlying clustering architecture in which the cluster head plays the key role in the network management. We propose to delegate the operations of handshake and key update defined above to cluster heads over the network which is going to minimize the overhead due to these two operations.

The handshake executed by each cluster head and the base station destined to establish a symmetric shared key between sensors and the base station. This handshake is executed in three steps:

Symmetric key generation: Each cluster head generates a random symmetric key, encrypts this key with the public key of the base station and sends it to the base station using the underlying routing protocol in an ordinary packet. The use of the public for transporting the session key ensures authentication, integrity and confidentiality of the handshake.

Establishment of the session key: After receiving and decrypting the message containing the session key coming from each cluster head, the base station stores all the keys in a global table used for identifying and managing clusters over the network.

Completion of the handshake: in order to finish and validate the handshake the base station generates a challenge message for each cluster head encrypted using the established session key. Each cluster head decrypts this message if this operation success the handshake is successfully completed otherwise the same operation is repeated until a valid session key is established.

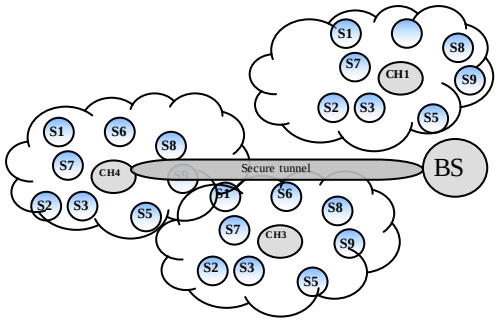

Fig. 1: Secure tunnel over hierarchical network

8.2 Distribution of the Session Key to Sensors

After a successful cluster head to base station handshake, each cluster member must get the same session key used by its cluster head. Therefore, each cluster member builds a message containing the identifier of its cluster head and a symmetric key used to secure the current operation, this message is encrypted using the public key of the base station.

When the base station receives this message, it seeks the existence of the corresponding session key of the cluster head (established during the previous handshake), encrypts it with the session key sent by the sensor and sends it to that sensor. Each sensor when receives this message from the base station, it shares the same session key with its cluster head and the base station which allow data aggregation and group security.

The dialogue is done with the base station instead of the cluster head because the base station is authenticated using its public key distributed before deployment.

8.3 System Functioning

Using the same mechanism defined for flat network to ensure data integrity, we propose to add a new field to the original structure of packet which contains MAC (message authentication code) encrypted with the session key established in the previous handshake.

The MAC can also be encrypted using the public key of the base station if sensors have the necessary resources to accomplish this operation.

8.4 Key Update

In order to protect the network from long term attacks, we propose to launch automatically a periodic key update. The key update is launched by the cluster head using the same handshake defined above in order to establish a new session key between the base station and the cluster head.

After updating the session key of the cluster head, each cluster head encrypts a copy with the old session key for each member of its cluster. The new session key will be automatically used after receiving the message by sensors in a given cluster.

IX. Energy Cost Analysis

9.1 Handshake Energy Cost

As mentioned above two types of cryptography are used in our proposed scheme, symmetric and asymmetric with optional use of a hash function as a MAC (message authentication code). We propose to use the ECC (Elliptic Curve Cryptography) for asymmetric encryption considered to be more effective regarding energy consumption. For symmetric encryption we propose to use AES (Advanced Encryption System) standardized as a tool for cryptography for future networks.

Taking into consideration the size of session keys (check sum, ID), the maximum packet size will not exceed the 512-bit so the consumed energy for its transmission is 3.78 mJ and 1.83 mJ for reception, using as platform Mica2dots [26], the energy consumed for data encryption and decryption is 22,82mj for asymmetric encryption and 0,039mj for symmetric encryption. Therefore the total energy consumption for each handshake in flat or hierarchical network is 28,47mj.

Table 2: Energy cost of handshake

Operations		Energy(mJ)
Base station to Sensor handshake	Encrypt session key	22,82
	Send session key	3,78
	Receive session key	1,83
	challenge message	0,039
Total energy cost		28,47

9.2 Key Update Energy Cost

As we have previously described a periodic key update is primordial for any security scheme in order to avoid long term attacks.

For flat networks, the key update is achieved by the execution of the same handshake for each sensor, which consumes the same amount of energy at each key update, for a network size of N; the consumed energy is N×28, 47 mJ.

However for hierarchical networks, it consumes less energy since the operation of the key update is delegated to cluster heads, for a network size of N sensors, the energy consumption for a cluster head key update is 28.47 mJ, assuming that 10% of sensors are cluster heads, the remaining nodes (90%) only need to decrypt the session key sent by the cluster head which consumes 0.039mj.

So the total consumption is:

$C = [(28,47 \times 10\%) + (0.039 \times 90\%)] N$

$N = 2,88 \times NmJ$

X. Security Analysis

In this section we try to analyze our proposed scheme regarding the guaranty of key management criteria during the network lifetime:

10.1 Security Services

Confidentiality: this aspect is ensured by the use of symmetric encryption to encrypt ordinary traffic between the base station and sensors. For more confidentiality we have enforced this mechanism using periodic key update to prevent long term attacks.

Authentication: this aspect is ensured by using public key cryptography by the base station; this public key is preloaded to each sensor before deployment which ensures the authentication of the base station using the corresponding private key as well as sensors, since only legitimate node has the valid public key preloaded before deployment.

Integrity: the integrity is ensured using the MAC (Message authentication codes) computed and joined to each packet, this MAC can also be encrypted using the public key of the base station which ensures more integrity and authentication.

10.2 Key Management Services

To be efficient and useful for all configurations and topologies of the network a key management scheme must ensure the following proprieties:

Availability: this propriety guaranties that the service of security is available during all the network lifetime, in our proposed scheme this is guaranteed since every entity over the network is responsible of its cryptographic keys as well as its link with the base station.

Fault Tolerance: this propriety deals with the continuation of services whenever one or more nodes over the network fail, this propriety is guaranteed since the loss of any node over the network does not affect the security service.

Scalability: this propriety deals with the network widening, in our proposed scheme the new coming sensors are managed by the execution of new handshakes and the creation of new clusters in the hierarchical architecture.

10.3 Resistance to attacks

As defined above the proposed key management scheme establish a set of session keys to secure traffic between the base station and each sensor over the networks. Using these session keys and the public key of the base station, it seems that the proposed scheme can resist against the main attacks:

Eavesdropping: this attack consists to passively listen to the exchanged data, in the proposed scheme this attack is avoided using symmetric encryption between each communicating entities enforced using an automatic key update.

Spoofing: the spoofing attack is avoided in our scheme by using the public key encryption, which guaranties the authenticity of the base station using the corresponding private key. In the other hands the authenticity of sensors is guaranteed using the same strategy since only legitimate nodes have the valid public key of the base station preloaded for sensors before deployment which ensures a mutual authentication between the base station and sensors.

Modification, ***reply and insertion***: These kinds of attacks alter the integrity of the exchanged data; these attacks are easily avoided by the symmetric encryption as well as message authentication code joined to each packet. So only authenticated nodes can insert or modify data over the network, and any other packet is rejected.

Avoiding other attacks : using the message authentication code and data encryption as well as the asymmetric encryption ensure both integrity, authentication and confidentiality which guaranties that falsified messages are automatically rejected from the network, in the way that only authenticated messages are forwarded which stop lot of attacks such as black hole attacks and denial of service attacks.

For more security, the proposed scheme can be used by routing protocols to secure and authenticate routing or to implement intrusion detection modules since the

proposed solution tries to make in practice a mechanism to ensure a secure key distribution.

XI. Conclusion

In this paper we have treated the aspect of security in wireless sensor networks, by proposing a key management scheme using public key cryptography for ensuring the authentication and the confidentiality key distribution using a set of handshakes based on the authenticity of the base station. The proposed key management is given in two versions according to the network architecture flat or hierarchical.

The first version is destined to flat network and uses one handshake to share a symmetric encrypting key with the base station and each sensor over the network, the established session key is used to encrypt ordinary traffic.

The second version of the proposed scheme treats the hierarchical architecture of wireless sensor networks in which the operation of handshake and key update are delegated to cluster heads which play the key role of the network security, this have considerably minimized the energy consumption compared to flat network version.

From the security point of view it seems that the proposed key management guaranties a great security threshold with a minimum of energy consumption.

Reference

[1] Hande Alemdar, Cem Ersoy, "Wireless sensor networks for healthcare: A survey", Computer NetworksVolume 54, Issue 15, pp. 2688-2710, 2010.

[2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey", Computer Networks, Vol. 38, No. 4, pp. 393–422.

[3] David Culler, Deborah Estrin, and Mani Srivastava, "Overview of Sensor Networks", IEEE Computer society, Vol. 37, No. 8, pp. 41-49, 2004.

[4] Carlos F.Garcia-hermandez and al, "Wireless sensor networks and applications", International Journal of Computer Science and Network Security, Vol.7, No.3, pp. 264-273, 2007.

[5] Gungor V.C., Bin Lu, Hancke G.P., "Opportunities and Challenges of Wireless Sensor Networks in Smart Grid", Industrial Electronics, IEEE Transactions, Vol. 57 No. 10, pp. 3557–3564.

[6] Karaboga, D.; Okdem, S.; Ozturk, C. "Cluster based wireless sensor network routings using Artificial Bee Colony Algorithm", international conference on Autonomous and Intelligent Systems (AIS), pp. 1 – 5, 2010

[7] T Kavitha, D Sridharan, "Security Vulnerabilities In Wireless Sensor Networks: A Survey", Journal of Information Assurance and Security (2010), Vol. 5, No. 1, pp. 31-44

[8] Syed Muhammad Khaliq-ur-RahmanRaazi and SungyoungLee ."A Survey on Key Management Strategies for Different Applications of Wireless Sensor Networks".Journal of Computing Science and Engineering, Vol. 4, No. 1, 2010.

[9] Du, W. ; Deng, J. ; Han, Y.S. &Varshney, P.K. "A pair-wise key pre-distribution scheme for wireless sensor networks". Proceedings of the 10th ACM Conference on Computer and Communications Security, pp. 42-51, 2003.

[10] Chan, H. ;Perrig, A. & Song, D. Random key pre-distribution schemes for sensor networks. Proceedings of the IEEE Symposium on Security and Privacy, p. 197, IEEE Computer Society Press, 2003.

[11] Xiao, y., v. k. rayi, b. sun, x. du, f. hu, and m. galloway. "A survey of key management schemes in wireless sensor networks". Computer Communications, Special issue on security on wireless ad hoc and sensor networks, pp 2314–2341, 2007.

[12] S. Zhu, S. Setia, and S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks", In 10th ACM conference on Computer and communication security, pp. 62-72, 2003.

[13] Basagni, Herrin, et al. "Secure pebblenets". Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing, 2001.

[14] C. Karlof, N. Sastry, and D.Wagner, "Tinysec A link layer security architecture for wireless sensor net-works," Second ACM Conference on Embedded Net-worked Sensor Systems, pp. 162-175, 2004.

[15] Watro, R. ; Kong, D. ; Cuti, S. ; Gardiner, C. ; Lynn, C. &Kruus, P. "TinyPK : securing sensor networks with public key technology". Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'04), pp. 59 – 64, 2004.

[16] Johann.G, Alexander.S, Stefan.T. "The Energy Cost of Cryptographic Key Establishment", in Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security, pp. 380–382, 2007.

[17] Jun Zheng and Abbas Jamalipour, "Wireless Sensor Networks: A Networking Perspective", a bookpublished by A John & Sons, Inc, and IEEEE, 2009.

[18] Shio Kumar Singh, M P Singh, and D K

Singh."Routing Protocols in Wireless Sensor Networks – A Survey ". International Journal of Computer Science & Engineering Survey, Vol.1, No.2, 2010.

[19] Deng, J. ; Han, R. & Mishra, S. "Countermeasures against traffic analysis in wireless sensor networks", Technical Report : CU-CS-987-04, University of Colorado at Boulder, 2004.

[20] Sen, J ; Chandra, M.G. ; Harihara, S.G. ; Reddy, H. &Balamuralidhar, P. "A mechanism for detection of grayhole attack in mobile ad hoc networks". Proceedings of the 6th International Conference on Information, Communication, and Signal Processing, pp. 1 – 5, 2007.

[21] Wood, A.D. &Stankvic, J.A. "Denial of service in sensor networks". IEEE Computer, Vol. 35, No. 10, pp. 54-62, 2002.

[22] Douceur, J. "The Sybil attack", Proceedings of the 1st International Workshop on Peer-to-Peer Systems, 2002.

[23] Perrig, A. ;Stankovic, J. & Wagner, D. "Security in wireless sensor networks". Communications of the ACM, Vol. 47, No. 6, pp. 53 – 57, 2004.

[24] N. Gura, A. Patel, A. Wander, H. Eberle, and S.C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs," Proceedings of the Sixth Workshop on Cryptographic Hardware and Embedded Systems, pp. 119-132, 2004.

[25] Crossbow Technology Inc., Processor/Radio Modules, 2008. (http://www.xbow.com).

Benamar Kadri is an associate professor in wireless network security, received his engineer degree in computer science in 2004, and his M.S. degree in 2006 from the University of Tlemcen, Algeria. Finished his PhD in wireless ad hoc networks security and routing in 2010. Member of STIC laboratory in the University of Tlemcen, his recent work is dealing with mobile wireless networks, their security, routing and management.

Mohammed Feham received his PhD in Engineering in optical and microwave communications from the University of Limoges, France in 1987, and his PhD in science from the university of Tlemcen, Algeria in 1996. Since 1987 he has been assistant professor and professor of microwave and communication engineering his research interest is in telecommunication systems and mobile networks.

Abdallah M'hamed is professor in Network security and dependability. He received his Doctor degree in dependability studies from the Technological University of Compiegne, France. In 1990 he joined the National Institute of Telecommunications, in Evry France. Member of the Handicom laboratory, his recent research activities are focused on authentication protocols and architectures, security and privacy in smart environments.

How to cite this paper: Benamar KADRI, Mohammed FEHAM, Abdellah MHAMMED,"Architecture Aware Key Management Scheme for Wireless Sensor Networks", IJITCS, vol.4, no.12, pp.50-59, 2012. DOI: 10.5815/ijitcs.2012.12.05

