Int. J. Advanced Networking and Applications
Volume: 02, Issue: 03, Pages: 707-713 (2010)

Analysis of Security Protocols in Wireless Sensor Network

Ritu Sharma[1]
Email: drrituji@gmail.com
Yogesh Chaba[2]
Associate Professor
Email : yogeshchaba@yahoo.com
Yudhvir Singh[3]
Assistant Professor
Email: yudhvirsingh@rediffmail.com
Department of Computer Science & Engineering [1,2,3]
Guru Jambheshwar University of Science & Technology, Hisar (Haryana) – India

ABSTRACT

Wireless Sensor Networks(WSNs) consists of low power, low-cost smart devices which have limited computing resources. With a widespread growth of the applications of WSN, the security mechanisms are also be a rising big issue. A lot of real-world applications have been already deployed and many of them will be based on wireless sensor networks. These applications include geographical monitoring, medical care, manufacturing, transportation, military operations, environmental monitoring, industrial machine monitoring, and surveillance systems. This paper discusses typical constraints, security goals, threat models and typical attacks on sensor networks and their defensive techniques or countermeasures relevant to the sensor networks, including security methods. The most critical area prone to attack is nearby the base station as the data is more aggregated, that should be kept secure using a number of defensive techniques as stated.

Keywords – Attacks, Threats, Security, Sensor nodes, Wireless Sensor Network.

Date of Submission: June 02, 2010 Revised July 07, 2010 Date of Submission: August 24, 2010

1. INTRODUCTION

WSN are composed of a large set (hundreds to a few thousand) of homogeneous nodes with extreme resource constraints. Each sensor node has wireless communication capability plus some level of intelligence for signal processing and data networking. These nodes are usually scattered over the area to be monitored to collect data, process it, and forward it to a central node for further processing. Military sensor networks might detect and gather information about enemy movements of people and equipment, or other phenomena of interest such as the presence of chemical, biological, nuclear, radiological, explosive materials. WSNs can support a myriad of uses including military, commercial, environmental, and medical applications. Natural environments such as remote ecosystems, disaster sites, endangered species, agriculture conditions, and forest fires can also be monitored with sensor networks[1].

Sensor networks are small, low-cost, low-power devices with the following functionality: they communicate over short distances, sense environmental data, and perform limited data processing. A typical node might have only 4MHz of processing power, 4KB of RAM, and a short transmission distance of less than 100 feet. Tiny OS is a small, open-source operating system developed to support most WSN applications. Wireless sensor networks often contain one or more sinks that provide centralized control. A sink typically serves as the access point for the user or as a gateway to another network. The sensor nodes communicate using RF, so broadcast is the fundamental communication primitive[2].

Security is one of the most difficult problems facing these networks. For certain applications of sensor networks, like military applications, security becomes very important. First, wireless communication is difficult to protect since it is realized over a broadcast medium. In a broadcast medium, adversaries can easily eavesdrop on, intercept, inject, and alter transmitted data. Second, since sensor networks may be deployed in a variety of physically insecure environments, adversaries can steal nodes, recover their cryptographic material, and pose as authorized nodes in the network. Third, Sensor networks are vulnerable to resource consumption attacks. Adversaries can repeatedly send packets to drain a node battery and waste network bandwidth. In these and other vital or security- sensitive deployments, secure transmission of sensitive digital information over the sensor

network is essential. The use of encryption or authentication primitives between two sensor devices requires an initial link key establishment process, which must satisfy the low power and low complexity requirements[3].

This paper focuses on different types of attacks and their defensive techniques within wireless sensor networks. Section 1 lists the introduction of wireless sensor network. Section 2 discusses constraints, security requirements, threat models, attacks for wireless sensor networks. Section 3 lists the Security solutions for establishing a secure sensor network. Finally article is concluded in Section 4.

2. WIRELESS SENSOR NETWORK

The most important security issues in WSN is its inherent security limitations. Before discussing the various threat models and various possible attacks and their countermeasures in a WSN, the basic security requirements or goals to achieve, is very much needed.

2.1 CONSTRAINTS IN WSN

Resource constraints: Sensor nodes have limited resources, including low computational capability, small memory, low wireless communication bandwidth, and a limited, usually no rechargeable battery.

Small message size: Messages in sensor networks usually have a small size compared with the existing networks. As a result, there is usually no concept of segmentation in most applications in WSN.

Addressing Schemes: Due to relatively large number of sensor nodes, it is not possible to build global addressing schemes for deployment of a large number of sensor nodes as overhead of identity maintenance is high.

Sensor location and redundancy of data: Position awareness of sensor network is important since data collection is normally based on location. Also there may be common phenomena to collect data, so there is a high probability that this data has some redundancy.

2.2 SECURITY REQUIREMENTS

The goal of security services in WSN is to protect the information and resources from attacks and misbehavior. The security requirements in WSN include:

a. Availability: Ensures that the desired network services are available even in the presence of denial of service attacks.
b. Authorization: Ensures that only authorized sensors can be involved in providing information to network services.
c. Authentication: Ensures that the communication from one node to another node is genuine. That is, a malicious node cannot masquerade as a trusted network node.
d. Confidentiality: Ensures that a given message cannot be understood by anyone other than the desired recipients.
e. Integrity: Ensures that a message sent from one node to another is not modified by malicious intermediate nodes.
f. Non-repudiation: Denotes that a node cannot deny sending a message it has previously sent.
g. Data Freshness: Implies that the data is recent and ensures that no adversary can replay old messages.
h. Robustness- When some nodes are compromised the entire network should not be compromised.
i. Self-organization-Nodes should be flexible enough to be self-organizing (autonomous) and self-healing (failure tolerant).
j. Time Synchronization- These protocols should not be manipulated to produce incorrect data.

2.3 THREAT MODELS

According to Karlof et. al. [4], threats in wireless sensor network can be classified into the following categories:

a. Outsider versus insider attacks: The outsider attacks regard attacks from nodes which do not belong to a WSN. An outsider attacker has no access to most cryptographic materials in sensor network. The insider attacks occur when legitimate nodes of a WSN behave in unintended or unauthorized ways. The inside attacker may have partial key material and the trust of other sensor nodes. Inside attacks are much harder to detect.
b. Passive versus active attacks: Passive attacks are in the nature of eavesdropping on, or monitoring of packets exchanged within a WSN; The active attacks involve some modifications of the data steam or the creation of a false stream in a WSN.
c. Mote-class versus laptop-class attacks: In mote-class attacks, an adversary attacks a WSN by using a few nodes with similar capabilities as that of network nodes. In laptop-class attacks, an adversary can use more powerful devices like laptop, etc. and can do much more harm to a network than a malicious sensor node.

2.4 ATTACKS

Attacks against wireless sensor networks are categorized as invasive or non-invasive. Non- invasive attacks generally consist of side channel attacks such as power, timing or frequency based attacks. There is not much work published about side channel attacks that target WSN specifically, but many of the problems found with other embedded systems, such as timing attacks against MAC generation or encryption, could be used against sensor nodes. Invasive attacks are much more common and the more important of these are described in the following sections. Several attacks on sensor networks are listed as follows:

Volume: 02, Issue: 03, Pages: 707-713 (2010)

A. Denial-of-Service(DoS) attack

In the denial-of-Service(DoS) attack, the hackers's objective is to render target machines inaccessible by legitimate users. There are two types of DoS attacks:
Passive attack: Selfish nodes use the network but do not cooperate, saving battery life for their own communications, they do not intend to directly damage other nodes.
Active attack: Malicious nodes damage other nodes by causing network outage by partitioning while saving battery life is not a priority.

Dos attacks can happen in multiple WSN protocols layers. At physical layer, the DoS attack could be jamming and tempering, at link layer, collision, exhaustion, unfairness, at network layer, neglect and greed, homing, misdirection, black holes and at transport layer, this attack could be performed by malicious flooding and desynchronization. The mechanisms to prevent DoS attacks include payment for network resources, pushback, strong authentication and identification of traffic.

B. Attacks on Information in Transit

The most common attacks against WSNs are on information in transit between nodes. Information in transit is vulnerable to eavesdropping, modification, injection, that can be prevented using well established confidentiality, authentication, integrity and replay protection protocols. Traffic analysis can potentially be a big problem in WSNs allowing an attacker to map the routing layout of a network, enabling very tightly targeted attacks to disrupt chosen portions of a network for greatest effect.

C. Node Replication Attack

A node replication attack involves an attacker inserting a new node into a network which has been cloned from an existing node, such cloning being a relatively simple task with current sensor node hardware. This new node can act exactly like the old node or it can have some extra behavior, such as transmitting information of interest directly to the attacker. A node replication attack is serious when the base station is cloned. However, as for many deployments, the base station is both in a secure location and much more powerful than the rest of the sensor nodes, so cloning it is much more difficult.

D. Routing attack

As with almost all networks there are a number of attacks that target the routing protocol of WSNs, all of which are necessarily insider attacks. Some are as follows:

a. Selective forwarding

Selective forwarding is a way to influence the network traffic by believing that all the participating nodes in network are reliable to forward the message. In selective forwarding attack, malicious nodes simply drop certain messages instead of forwarding every message. Malicious or attacking nodes can refuse to route certain messages and drop them. If they drop all the packets through them, then it is called a blackhole attack. However, if they selectively forward the packets, then it is called selective forwarding. Effectiveness of this attack depends on two factors. First the location of the malicious node, the closer it is to the base station the more traffic it will attract. Second is the percentage of messages it drops. When selective forwarder drops more messages and forwards less, it retains its energy level thus remaining powerful to trick the neighboring nodes.

b. Sinkhole attacks

In sinkhole attacks, adversary attracts the traffic to a compromised node. The simplest way of creating sinkhole is to place a malicious node where it can attract most of the traffic, possibly closer to the base station or malicious node itself deceiving as a base station. One reason for sinkhole attacks is to make selective forwarding possible to attract the traffic towards a compromised node. The nature of sensor networks where all the traffic flows towards one base station makes this type of attacks more susceptible.

c. Sybil attacks

In Sybil attack, a single node presents multiple identities to all other nodes in the WSN. This may mislead other nodes, and hence routes believed to be disjoint w.r.t node can have the same adversary node. Sybil attacks can be used against routing algorithms and topology maintenance; it reduces the effectiveness of fault tolerant schemes such as distributed storage and dispersity. Another malicious factor is geographic routing where a Sybil node can appear at more than one place simultaneously.

d. Wormholes

In wormhole attacks, an adversary positioned closer to the base station can completely disrupt the traffic by tunneling messages over a low latency link. Here an adversary convinces the nodes which are multi hop away that they are closer to the base station. This creates a sinkhole because adversary on the other side of the sinkhole provides a better route to the base station.

e. Flooding

Sometime, the malicious node can cause immense traffic of useless messages on the network. This is known as the flooding. Sometimes, malicious nodes replay some actual broadcast messages, and hence generating useless traffic on the network. This can cause congestion, and may eventually lead to the exhaustion of complete nodes. This is a form of Denial of Service attack.

Security in wireless sensor networks is a critical issue keeping in view limitations and application domains of sensor networks. In sensor networks there is need to maintain a delicate balance between security and network operations. The techniques such as Link Layer encryption

and authentication, multipath routing, identity verification and authenticated broadcast seem to be good solution for security in WSN. However attacks such as Sinkhole and Wormholes pose lot of challenges to secure routing protocol design. Geographical Routing Protocols is one example of routing protocols which are able to withstand most of the WSN routing based attacks, as the legitimate nodes are able to estimate the location of the adversary nodes. Hence attacks such as Sybil are effective. Effective and Efficient countermeasures are still lacking against these attacks, which can be applied after the design of these routing protocols has completed. So there exist a severe need to design such routing protocols in which these attacks are ineffective.

3. SECURITY SOLUTIONS IN SENSOR NETWORKS

Security schemes can be applied to provide security in wireless sensor networks, but keeping in view their resource starved nature it is very difficult to do so. Some researchers are striving to develop improved WSN protocols, others are attempting to improve node design; still others are working to resolve security issues including the main WSN security threat of insecure radio links with eavesdropping and information corruption possible. Most security mechanisms that exist today require intensive computation and memory which is the limiting factor in wireless sensor networks. Many security mechanisms require repeated transmission/communications between the sensor nodes which are further drawn in their resources. The number of security suites already exist that are at least some way appropriate for use in WSNs, and combat some of the threats to these networks. This section review some of the more popular and more suitable solutions here.

3.1 SPINS: Security Protocols For Sensor Networks

Adrian Perrig *et al.*[5] proposed “SPINS” a suite of security protocols optimized for sensor networks. SPINS has two secure building blocks: SNEP and µTESLA. SNEP includes: data confidentiality, two-party data authentication, and evidence of data freshness. µTESLA provides authenticated broadcast for severely resource-constrained environments.

3.1.1 SNEP: Sensor Network Encryption Protocol

SNEP provides a number of following advantages.

1. It has low communication overhead as it only adds 8 bytes per message.
2. Like many cryptographic protocols it uses a counter, but avoids transmitting the counter value by keeping state at both end points.
3. SNEP achieves semantic security, which prevents eavesdroppers from inferring the message content from the encrypted message.
4. Finally, SNEP protocol offers data authentication, replay protection, and weak message freshness.

However, sending data over the RF channel requires more energy. So, SNEP construct another cryptographic mechanism that achieves semantic security with no additional transmission overhead. It relies on a shared counter between the sender and the receiver for the block cipher in counter mode (CTR). Since the communicating parties share the counter and increment it after each block, the counter does not need to be sent with the message. To achieve two-party authentication and data integrity, SNEP uses a message authentication code (MAC).The combination of these mechanisms form Sensor Network Encryption Protocol SNEP.

SNEP offers the following properties:

- Semantic security: Since the counter value is incremented after each message, the same message is encrypted differently each time. The counter value is long enough that it never repeats within the lifetime of the node.
- Data authentication: If the MAC verifies correctly, the receiver can be assured that the message originated from the claimed sender.
- Replay protection: The counter value in the MAC prevents replaying old messages. Note that if the counter were not present in the MAC, an adversary could easily replay messages.
- Weak freshness: If the message verified correctly, the receiver knows that the message must have been sent after the previous message it received correctly (that had a lower counter value). This enforces a message ordering and yields weak freshness.
- Low communication overhead: The counter state is kept at each end point and does not need to be sent in each message.

3.1.2 µTesla: Authenticated Broadcast

Asymmetric digital signatures are impractical for sensor networks for the authentication, as they require long signatures with high communication overhead of 50-1000. Earlier TESLA protocol provided efficient authenticated broadcast However, TESLA was not designed for sensor networks. Adrian Perrig et al. proposed µTESLA to solve the following inadequacies of TESLA in sensor networks:

- TESLA authenticates the initial packet with a digital signature, which is too expensive for our sensor nodes. µTESLA uses only symmetric mechanisms.
- Disclosing a key in each packet requires too much energy for sending and receiving. µTESLA discloses the key once per epoch.
- It is expensive to store a one-way key chain in a sensor node. µTESLA restricts the umber of authenticated senders.

µTESLA OVERVIEW

The basic idea of the µTesla system is to achieve asymmetric

cryptography by delaying the disclosure of the symmetric keys. In this case a sender will broadcast a message generated with a secret key. After a certain period of time, the sender will disclose the secret key. The receiver is responsible for buffering the packet until the secret key has been disclosed. After disclosure the receiver can authenticate the packet, provided that the packet was received before the key was disclosed. One limitation of µTesla is that some initial information must be unicast to each sensor node before authentication of broadcast messages can begin.

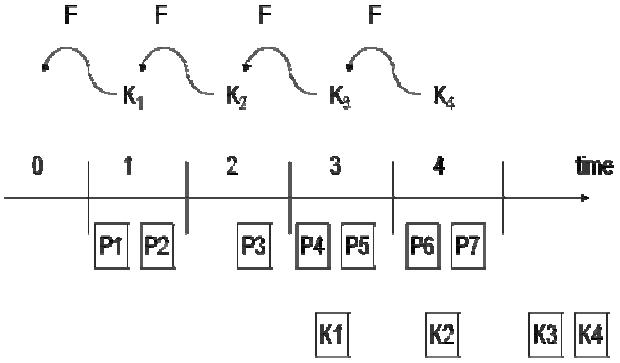

Figure 3.1 An example of µTESLA.

Assume that the receiver node is loosely time synchronized and knows K0 (a commitment to the key chain) in an authenticated way. As the fig. 3.1 shows that Packets P1 and P2 sent in interval 1 contain a MAC with key K1. Packet P3 has a MAC using key K2. So far, the receiver cannot authenticate any packets yet. Let us assume that packets P4, P5, and P6 are all lost, as well as the packet that discloses key K1, so the receiver can still not authenticate P1, P2, or P3. In interval 4 the base station broadcasts key K2, which the node authenticates by verifying K0 = F (F(K2)), and hence knows also K1 = F(K2), so it can authenticate packets P1, P2 with K1, and P3 with K2. Instead of adding a disclosed key to each data packet, the key disclosure is independent from the packets broadcast, and is tied to time intervals. Within the context of µTESLA, the sender broadcasts the current key periodically in a special packet.

3.2 TINYSEC

Karlof *et al*. designed the replacement for the unfinished SNEP, known as TinySec[6]. Inherently it provides similar services, including authentication, message integrity, confidentiality and replay protection. A major difference between TinySec and SNEP is that there are no counters used in TinySec. For encryption, it uses CBC mode with cipher text stealing , and for authentication, CBC-MAC is used. TinySec XORs the encryption of the message length with the first plaintext block in order to make the CBC-MAC secure for variably sized messages. There are two packet formats defined by TinySec. These are TinySec-Auth, for authenticated messages, and TinySec-AE, for authenticated and encrypted messages. For the TinySec-AE packet, a payload of up to 29 Bytes is specified, with a packet header of 8 Bytes in length. Encryption of the payload is all that is necessary, but the MAC is computed over the payload and the header. The TinySec- Auth packet can carry up to 29 Bytes of payload. The MAC is computed over the payload and the packet header, which is 4 Bytes long. Generally, the security of CBC-MAC is directly related to the length of the MAC. TinySec specifies a MAC of 4 Bytes, much less than the conventional 8 or 16 Bytes of previous security protocols. In the context of sensor networks, Karlof *et al.* argue that this is not detrimental . Should an adversary repeatedly attempt blind forgeries, it will succeed after 2^{31} attempts. adversaries can only assess the validity of an attempted forgery by forwarding it to an authorised recipient. This implies that approximately 2^{31} packets must be sent to forge just one malicious packet. In sensor networks, this is an adequate level of security, and for an attempt like the one described above, it would take approximately 20 months (on a 19.2kb/s channel) to be successful. Implicitly, there is an effective denial of service attack launched in this way, as the radio channel would be locked for an extended period as attempts are made. It is argued that a simple heuristic, whereby the nodes signal the base station when the rate of MAC failures exceeds a predetermined threshold would alleviate the problem should such an attack occur.

3.3. MINISEC

MiniSec [7] is a secure network layer protocol that claims to have lower energy consumption than TinySec while achieving a level of security which matches that of Zigbee. A major feature of MiniSec is that it uses offset codebook (OCB) mode as its block cipher mode of operation, which offers authenticated encryption with only one pass over the message data. Normally two passes are required for both secrecy and authentication. Another major benefit of using OCB mode is that the ciphertext is the same length as the plaintext, disregarding the additional fixed length tag, four bytes in MiniSec's case, so padding or ciphertext stealing is not necessary. Another primary feature MiniSec has over the other security suites mentioned here is strong replay protection without the transmission overhead of sending a large counter with each packet or the problems associated with synchronized counters if packets are dropped. To achieve this MiniSec has two modes of operation, one for unicast packets MiniSec-U, and one for broadcast packets.

3.4 LEAP: Localized Encryption And Authentication Protocol

Sencun Zhu *et. al*.[8] proposed LEAP Protocol, which is a key management protocol for sensor networks. LEAP is designed to support secure communications in sensor networks; therefore, it provides the basic security services such as confidentiality and authentication. In addition, LEAP

is to meet several security and performance requirements that are considerably more challenging to sensor networks.
LEAP has the following properties:
• LEAP assumes that no single keying mechanism is appropriate for all the secure communications that are needed in sensor networks. As such, LEAP supports the establishment of four types of keys for each sensor node – an individual key shared with the base station, a pairwise key shared with another sensor node, a cluster key shared with multiple neighboring nodes, and a group key that is shared by all the nodes in the network.
• LEAP includes an efficient protocol for local broadcast authentication based on the use of one-way key chains.

• A distinguishing feature of LEAP is that its key sharing approach supports in-network processing, while restricting the security impact of a node compromise to the immediate network neighborhood of the compromised node.
LEAP can prevent or increase the difficulty of launching many security attacks on sensor networks. The key establishment and key updating procedures used by LEAP are efficient and the storage requirements per node are small. LEAP is feasible for the current generation sensor nodes.

3.5 ZIGBEE

Zigbee[9] Coordinator acts as "Trust Manager", which allows other devices to join the network and also distributes the keys. It plays the three roles as follows : 1:Trust manager, whereby authentication of devices requesting to join the network is done, 2:Network manager, maintaining and distributing network keys, and 3:Configuration manager, enabling end-to-end security between devices. It operates in both Residential Mode and Commercial Mode. The Trust Center running Residential Mode is used for low security residential applications. Commercial Mode is designed for high-security commercial applications.

There are three types of keys employed, Master Key, Link Key and Network Key. Master keys are installed first, either in the factory or out of band. They are sent from the Trust Center and are the basis for long-term security between two devices. The Link key is a basis of security between two devices and the Network keys are the basis of security across the entire network. Link and Network keys, which are either installed in the factory or out of band, employ symmetrical key-key exchange (SKKE) handshake between devices. The key is transported from the Trust Center for both types of keys. This operation occurs in commercial mode, as residential mode does not allow for authentication.

3.6 802.15.4

The 802.15.4 standard[10] provides link layer security services, and has three modes of operation, unsecured, an Access Control List (ACL) mode and secured mode. In unsecured mode, as the name implies, no security services are provided. In ACL mode the device maintains a list of devices with which it can communicate. Any communication from devices not on the list is ignored. However, it must be noted that this mode offers no cryptographic security so it is trivial for the message source address to be spoofed. Secured mode offers seven security suites and depending on which is used any of four security services are offered, these being access control, data encryption, frame integrity and sequential freshness. One cryptographic algorithm, AES-128, is employed for all security suites, which allows for a very small implementation. For high security the full 128-bit message integrity code (MIC) can be added to each transmitted message but the MIC can be truncated to 64 or 32 bits to trade security for shorter message length.

802.15.4 security suites should be implemented on the radio chips all the necessary cryptographic computations are performed in hardware and reduces energy consumption. Some problems were found with security modes at the lower levels but higher level protocols overcome these limitations. Hence, 802.15.4 standard, if implemented correctly, can be used as a good base for building higher level, fully featured security suites.

4. CONCLUSION AND FUTURE WORK

Each of the authentication mechanisms are to be examined in a simulated environment and evaluated under the headings speed of operation, power consumption, efficiency and security level offered. The details for these mechanisms are available in section 3 and in addition a comparison table is given in the Table 4.1 of this paper. This is to further evaluate the effectiveness of these protocols and define their more desirable characteristics. There is currently no one solution that can be plugged-in to an application to provide all the necessary. The future goal of this research is to develop a new authentication protocol, by combining the most desirable traits of what currently exists and implementing some new ideas, which is optimal for implementation in wireless sensor network application security primitives.

	SPINS	LEAP	TINYSEC	ZIGBEE	802.15.4	MINISEC
Encryption	Yes	Yes	Yes	Yes	Yes	Yes
Freshness (CTR)	Yes	No	No	Yes	Yes	Yes
Overhead (Bytes)	8	Variable	4	4,8or 16	4,8or 16	4 + 3
MAC Used	Yes	Yes	Yes	Yes	Yes	Yes
Key Agreement	Symmetric Delayed	Pre-Deployed	ANY	Trust Center	-	ANY

Table 4.1 Comparision Table

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: A survey”, Computer Networks Journal, Elsevier Science, Vol. 38, No. 4, pp 393–422, March 2002.

[2] J. M. Kahn, R. H. Katz, and K.S. Pistcr, Mobile Networking for Smart Dust, ACM/IEEE International Conference on Mobile Computing (MobiCom ‘99), Seattle, WA, 1999, 217 – 278.

[3] J. Staddon, D. Balfanz, and G. Durfee. “Efficient tracing of failed nodes in sensor networks”, Proc. of the first ACM International workshop on Wireless sensor networks and applications (WSNA), ACM Press, 2002, 122-130.

[4] Chris Karlof and David Wagner, “Secure routing in wireless sensor networks: attacks and countermeasures”, Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols. 2003.

[5] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. In Seventh Annual ACM International Conference on Mobile Computing and Networks (MobiCom 2001), July 2001.

[6] C. Karlof, N. Sastry, and D. Wagner, "TinySec: a link layer security architecture for wireless sensor networks," in 2nd international conference on Embedded networked sensor systems, Baltimore, MD, USA, 2004, 162 – 175.

[7] M. Luk, G. Mezzour, A. Perrig, and V.Gligor, "MiniSec: A Secure Sensor Network Communication Architecture," in IEEE International Conference on Information Processing in Sensor Networks (IPSN'07), Cambridge, Massachusetts, USA, 2007.

[8] S. Zhu, S. Setia, and S. Jajodia. “Leap: efficient security mechanisms for largescale distributed sensor networks”, In CCS ’03: Proceedings of the 10th ACM conference on Computer and communications security, New York, USA, 2003, 62–72.

[9] ZigBee Specification v1.0: ZigBee Specification (2005), San Ramon, CA, USA: ZigBee Alliance. http://www.zigbee.org/en/spec_download/download_request.asp

[10] 802.15.4: Wireless Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LRWPANs), 2003 New York: IEEE Standards Association.