A Security Scheme for Wireless Sensor Networks

Hacène Fouchal , Javier Biesa , Elena Romero , Alvaro Araujo , Octavio Nieto Taladrez

Abstract

Security is critical for wireless sensor networks (WSN)deployed in hostile environments since many types of attacks could reduce the trust on the global functioning of any WSN. Many solutions have been proposed to secure communications for WSNs and most of them rely on a centralized component which behaves as a certificate authority. We propose in this paper a distributed solution able to ensure authentication of nodes at any time without having any on-line access to a certificate authority. Each node will be equipped with a Trusted Platform Module (TPM) which is able to store keys with security. Each node will have its own public key and private key pair in the TPM and a certificate of the public key. The certificate is issued off-line when setting-up the node. When a node communicates with another, it has to sign the message with its own private key (done securely by the TPM) and sends the message, the signature and the certificate of the public key. The evaluation of the solution has been done using simulation and the overhead added by integrating authentication does not exceed 15% of energy consumption.

***Index Terms*—WSN, Security, TPM, authentication.**

I. Introduction

Wireless sensor networks are based on a combination of sensing function with computing and communication which opens the possibility to many applications to be developed in many areas. Securing wireless sensor networks, is particularly challenging since many constraints exist:

- Reduced memory resources: we do not have enough space to store data about all nodes.
- Their ad-hoc nature, potentially forcing sensor nodes to interact with many different networks over time.
- Sensor nodes are not tamper resistant, therefore any secrete key or code could be extracted easily.
- The use of PKI is not possible since a connection to the internet (and to an certificate authority) is not possible.

Wireless sensor nodes have to work using their limited energy capacity for sensing, computing and transmitting information in a wireless environment since recharging or replacing batteries is not always possible.

Security is critical for sensor networks deployed in hostile environments. Indeed, it is well known that wireless communication is subjected to threats such as flooding attack, denial of service attack jamming attack, selective forwarding attack. For these reasons, ensuring node authentication protects communications against these attacks. Security mechanisms deployed in WSNs should involve collaboration between all the nodes due to the decentralized nature of the nodes and the absence of any infrastructure. In addition, it is much critical when the nodes are equipped with security items as secrete keys and any other sensitive data. Intruders may deploy among nodes some fake nodes similar to the actual nodes.

We propose in this paper an authentication method of transmitting data and a secure method to store sensitive data in a WSN thanks to a Trusted Platform Module (TPM). A TPM is used to ensure a secure storing of all required data to authenticate the nodes.

The paper is organized as follows. Section II is dedicated to related works. Section III details our security model. Section IV explains the model is implemented. Finally, section V gives a conclusion and some hints about future works.

II. Related works

Resources are very limited on WSNs nodes. For this main reason, all research studies on WSN have to consider carefully this issue. Indeed, if we intend to integrate security in a WSN, we should take care of this issue. Many research studies have proposed adapted security solutions for WSNs. In [1], an interesting survey is proposed about some security solutions on WSNs done before 2009. In [2], the concept is a TPM is presented with its architecture and all required components. [3] and [4] proposed a secure techniques on data aggregation, they focus on achieving the aggregation in a secure manner In [5] and [6], authors have proposed security solutions based on TPM. In [7], an authentication mechanism is proposed over WSN where nodes are equipped with TPM. But in this study, a certificate authority is implemented on a specific sensor.

III. The model

A. Basic cryptography framework

Public-key algorithms are mainly based on mathematical problems which are difficult to solve. Indeed, they consider some fundamental issues as integer factorization, discrete logarithm, and in some other cases elliptic curve relationships.

It is quite obvious that it is computationally easy for a user to generate a public and private key-pair. These keys could be used to ensure confidentiality or authentication. But is is well known that it is computationally very hard or impossible to generate a private key from its corresponding public key. Message confidentiality necessitate to encrypt a message with the public key and a decryption with the private one. Message authentication involves processing a message with a private key to produce a digital signature. Then any receiver can verify this signature by processing the signature value with the signer's corresponding public key and comparing that result with the received message. If this signatures matches then the message is authenticated.

Public-key algorithms are fundamental ingredients in security applications and protocols. Some public key algorithms provide distribution of keys for example Diffie"Hellman key exchange, some other algorithms provide digital signatures as DSA (Digital Signature Algorithm), and sometimes some others provide both as RSA.

Elliptic curve cryptography (ECC) is a public-key crypto-system similar to RSA, or El Gamal Algorithm. An elliptic curve is a plane figure defined by an equation having a shape like $y^2 = x^3 + ax + b$. The F(x, y)=0 (with 2 variables) gives a curve in the plane. We use geometric arithmetic methods to find the equation solutions.

Elliptic curves are used as an extension to other current crypto-systems like Elliptic Curve Diffie-Hellman Key exchange (ECDHK), Elliptic Curve Digital Signature algorithm (ECDSA). The use of such algorithms, participants should agree on some publicly-known data items like the elliptic curve equation, the values of a and b, prime p.

Then, an elliptic group (computed from the elliptic curve equation) is generated. A base point B, taken from the elliptic group Similar to the generator used in current crypto systems. Each user generates its public/private key pair where private key is an integer, x, selected from the interval [1, p-1] and the Public Key = product, Q, of private key and base point (Q = x*B). Elliptic curve cryptography is much faster and requires less memory therefore it is really suitable for WSN. For this reason, our proposed method is based a on ECDSA mechanism.

B. Trusted Platform Module

In our model for sensors will use mechanisms offered by a Trusted Platform Module (TPM). We consider that each node of the WSN is equipped with a TPM.

A TPM is an implementation of a standard developed by the Trusted Computing Group [8]. This module is designed to support various procedures and protocols that can be used for securing data. Mainly, the following functions are usually provided:

- generating an asymmetric key pair
- secure storage of keys,
- generating an electronic signatures,
- encryption and decryption,

C. Initialization step

Each node s_i should have its own public pub_i and private $priv_i$ key pair We consider in our proposal that a certificate authority (CA) works off-line for one main purpose: it will sign the public key pub_i of each node. Thereafter it will issue a certificate to each node. This operation is done once when a sensor is built. That means each node will have its own certificate and the CA public key stored securely.

D. Authenticated communication step

Our main contribution is proposed in this part. Indeed, if a node s_i needs to send a message M to another node s_j, s_i will compute a signature of this message thru its private key and will produce $Sign_i(M)$. Then s_i will send the message M, its signature $Sign_i(M)$ and its own certificate $Cert_i$. When s_j receives the message M' with the signature Sig'_1 and a certificate $Cert_i$, it will first extract s_i public key pub_i from $Cert_i$. It will then extract the sent signature from Sig_i which is compared the calculated signature on the received message. If both signatures are equal, then the message is authenticated. This process could be optimized in the sense that a node can send messages with their signatures and sometimes they insert the node certificate. The receiving node will keep this signed messages until they receive the certificate. At that moment, all recorded signed messages will be verified thanks to the certificate. This optimization could only be applied for non-real time applications.

IV. Evaluation of the model

In this section, we will detail the evaluation of our contribution through simulations

A. Simulator environment

The scenario has been implemented in Castalia simulator. The decision about the simulator is made following these reasons:

- Castalia simulator is focused on WSN. That means, Castalia includes several necessary features for our scenario.
- Castalia simulator is based on OMNET++. OMNET++ [9] proposes a modular library which could be used to develop network simulators. This simulator can support MAC protocols as well as some protocols used in WSN. In addition, OMNET++ has a mobility framework, and can monitor power consumption.
- Castalia physical layer and radio models are one of the most realistic models among the WSN simulators. Castalia offers multiple characteristics as path loss, mobility in the nodes, interference models, multiple modulations and node sleep states.
- Castalia includes some typical radio interfaces for WSNs, such as CC1010, CC2420 or CC2430, but it also accepts the implementation of new ones.
- According to the OMNET++, Castalia uses modules and messages to implement a WSN. Castalia nodes connect to each other through the wireless channel module which is responsible for delivering the messages. In addition, a physical process module implements the events simulation that occurs in the scenario.
- The node is a composite module. Each node contains a communication module (radio, MAC and routing layers) connected to the wireless module. The application layer is plugged on the communication module . Finally, each node has a resource manager that gathers information about the memory, energy, CPU consumption from every module in the node.

B. Scenario description

The simulated scenario is composed of 21 WSN nodes deployed in a 50 m x 50 m area. The nodes communicate over a star topology including 1 network coordinator and 20 end-devices (environment monitoring sensors). The coordinator position is fixed in the center of area, while the end-devices are randomly deployed following a uniform distribution. The simulated scenario uses the standard IEEE 802.15.4 over the 2.4 GHz ISM band.

For the simulation scenario, typical WSN packets are chosen. End-devices transmit WSN packets to the network coordinator of 4, 20 and 100 Bytes length at -5 dBm. Different data rates are simulated in order to evaluate their influence on the results. Different packet rates see selected from 1 to 20 packet/s for the different messages sizes.

WSN nodes are modeled by a Texas Instrument CC2420 transceiver commonly used for WSNs. A maximum number of 5 retransmissions are selected for the WSN nodes. The values of energy consumption are extracted from the CC2420 datasheet (for transmission, reception, idle modes and energy costs of transitions between modes) and verified through experimentation in a real testbed. Typical current consumptions of WSN nodes are 20 mA in transmission or reception mode and below 1 mA in a stand-by mode.

The use of the TPM and the certification described in the paper is introduced into the simulator by their cost in terms in energy consumption and the increase in packet length. As described in the datasheet, energy consumption at the TPM module is assumed 25 mA in active mode and 0.9 mA in idle mode. The signature is assumed as 28 bytes and the certificate is modeled as 64+28 bytes.

Different levels of authentication have been simulated in order to extract conclusions about the overhead introduced by the use of the TPM module and the authentication mechanism described below. Also, a basic scenario without authentication protocol is simulated for comparison purposes. All the results presented in this work show the energy consumption (in Joules) per second for the whole network of 20 nodes.

C. Energy consumption evaluation

In this section, we will show the results of our evaluation. We have undertaken various simulation situations. We have conducted some simulations with full authentication, that means each message will be sent with its own signature and its certificate. Some others have been achivied with 1 certificate sent for 5 messages sent. Some other situations we will not use authentication when we mention no-authentication. On the figures, we observe in all cases that there is an overhead due to authentication. It could be measured as 15% maximum in worst cases.

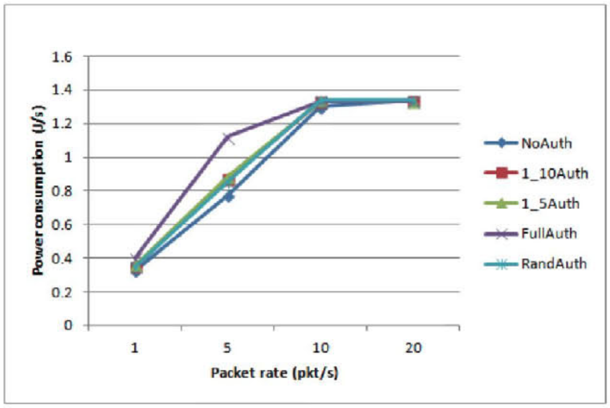

Figure 1. Energy consumed with various authenticated messages where message size is 4bytes

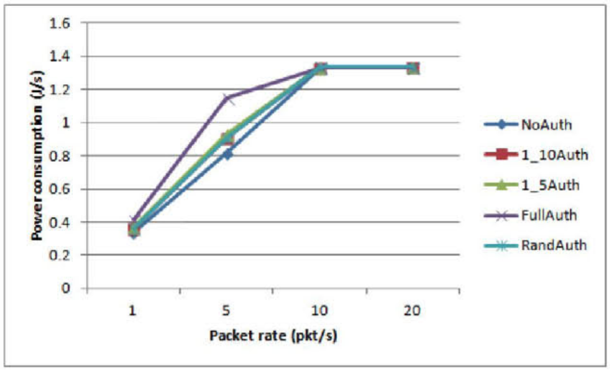

Figure 2. Energy consumed with various authenticated messages where message size is 20bytes

In the three figures Figure 1, Figure 2, and Figure 3, we observe that the energy consumed with authentication and without authentication differ with few joules. The main conclusion is that such a mechanism is simple to implement and does not consume high energy. In Figure 4, we summarize our results for various data rates. We observe that when the data rate increases, the influence of the authentication on the

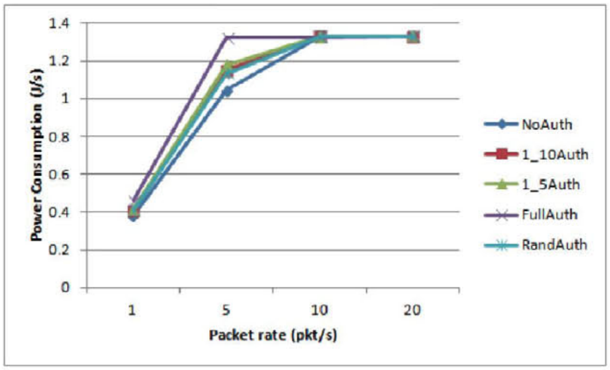

Figure 3. Energy consumed with various authenticated messages where message size is 100bytes

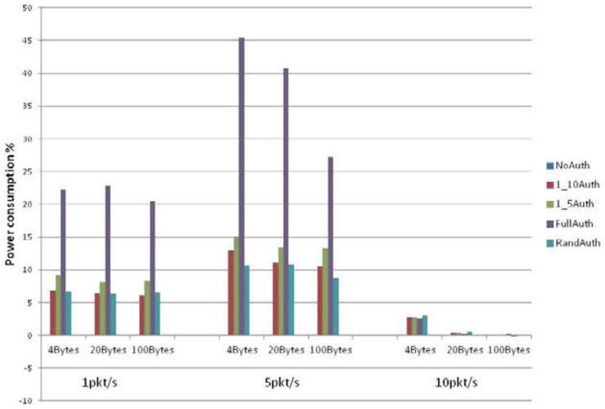

Figure 4. The overhead rate of energy when authentication is added

energy consumption is quite low. This is explained by the fact that with higher data rates, retransmission rate may be higher and the added consumed energy due to authentication is neglected compared to the consumed energy without authentication.

V. Conclusion & Future Works

We have proposed a simple and very realistic method able to authenticate nodes over a WSN. The method is robust since the nodes are equipped with a Trusted Plateform Module which will store with high security all keys. Each node is also certified by an off-line adhoc certificate authority (CA). Then each node will have its own public and private keys and a certificate from CA. When a node needs to authenticate its sent message, it has to send the message, a signature produced by the TPM and its own certificate. The receiver will be able to check the signature of the message since it is able to extract the sender's public key from the received certificate. Thereafter it will be able to verify the message's signature. The method is easy to implement and the obtained results seems very interesting. Indeed, integrating security does not cost as much as we could expect. The consumed energy does not exceed 15%. As a future work, we intend to study large scale WSN and check the generated overhead. Some optimizations could be investigated as reducing the certificate to send. They only could be sent when they are required.

References

[1] J. Sen, "A Survey on Wireless Sensor Network Security", *International Journal of Communication Networks and Information Security*, Vol. 1, No. 2, August 2009.

[2] S. Kinney, "Trusted platform module basics: using TPM in embedded systems", *Embedded Technology Series*, Elsevier Inc., 2006

[3] Y Mohamedd Yussoff, H. Hashim, M. Dani Baba, "Identity-based Trusted Authentication in Wireless Sensor Network", *International Journal of Computer Science Issues*, Vol. 9, Issue 3, No 2, May 2012.

[4] B. Przydatek, D. Song, and A. Perrig, "SIA: Secure Information Aggregation in Sensor Networks," *Proc. 1st Int'l. Conf. Embedded Networked Sensor Systems, SenSys '03,* New York: ACM Press, 2003, pp. 255-265,

[5] W. Hu, H. Tan, P. Corke, W. Chan Shih, S. Jha, "Toward Trusted Wireless Sensor Networks", *ACM Transactions on Sensor Networks*, Vol. 7, No. 1, Article 5, August 2010.

[6] C. KrauSS, F. Stumpf, C. Eckert, "Detecting Node Compromise in Hybrid Wireless Sensor Networks Using Attestation Techniques", *Lecture Notes in Computer Science Volume 4572*, Springer 2007, pp. 203-217.

[7] Janusz Furtak and Jan Chudzikiewicz "Securing transmissions between nodes of WSN using TPM" *the Federated Conference on Computer Science and Information Systems*, Lodz, Poland, September 2015, pp. 1071?1080

[8] TPM Main Part 1 Design Principles. Specification Version 1.2.Revision 116, Trusted Computing Group, Incorporated, 2011

[9] Xian, X.; Shi, W.; Huang, H. "Comparison of OMNET++ and other simulator for WSN simulation". *3rd IEEE Conference on Industrial Electronics and Applications*, 2008. ICIEA 2008. , 2008, pp. 1439 ?1443.