***I.J. Wireless and Microwave Technologies*, 2011, 6, 52-58**
Published Online December 2011 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijwmt.2011.06.08

Available online at http://www.mecs-press.net/ijwmt

Analysis of Trusted Identity Based Encryption (IBE-Trust) Protocol for Wireless Sensor Networks

Yusnani Mohd Yussoff[a], Habibah Hashim[b]

Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Selangor Malaysia

Abstract

The peculiarity of Wireless Sensor Networks demands extra consideration during the design of the security protocol. One of the most challenging yet important security features in Wireless Sensor Network is in establishing trusted communication between sensor node and base station. While the term trusted has been widely used referring to valid nodes in the group, this paper discuss the term trusted based on the specifications of Trusted Computing Group (TCG) and presents an IBE-Trust security protocol utilizing well-known identity based encryption scheme. The protocol incorporates ideas from Trusted Computing Group and Identity-based cryptosystem by Boneh Franklin in ensuring trusted and secured communications between sender and receiver. The proposed protocols were then modeled using the high-level formal language HLPSL and verified using the model checking tool AVISPA. Analysis on the proposed protocols is presented at the end of this paper.

Index Terms: Formal Analysis, WSN, Trusted Computing, Security

1. Introduction

Based on the potential threats in WSNs and the growth of applications related to WSNs, we believe that the security in WSNs should be considered at the very beginning of the development which is the design of the physical sensor. Furthermore, due to the eccentric characteristic of WSNs, the designer should also consider the following aspects in their security design; Constraints faced by the sensor node such as limited power, computational capabilities and memory size; High accessibility of nodes by anybody thus exposing them to intruders and attackers; Information exchanges are easily accessible due to public communication channel and finally difficulties to monitor the condition of the deployed sensor nodes which may be due to its location, mobility and environment.

While majority of the work done in WSN security have focused on the security of the network [1], our proposed works not only consider the physical design of the sensor node to protect important keys but also software aspects and the design aspects in establishing trusted communication between sensor node and base

* Corresponding author.
E-mail address: [a]yusna233@salam.uitm.edu.my; [b]habib350@salam.uitm.edu.my

station with minimum communication overhead [2]. While the first induce extra efforts to thwart physical attacks the second will ensure authenticity, confidentiality and integrity of the data and networks. Besides, our proposed scheme guarantees only trusted sensor nodes can exist in the network.

At this stage, the authors believe that embedding the security parameters in the processor is the most suitable technique for securing wireless sensor node. This technique is believed to be capable of reducing the size of the sensor node, decreasing the processing time and preventing software and physical attacks as well as providing other benefits. Johann et al. in his paper [3] also conclude that hardware based security features need to be integrated into the processor to avoid vulnerabilities such as those which exist in today’s personal computer. Besides secure implementation, the node also should communicate in a trusted environment. Tiago and Don [4] mentioned that the demand in trusted computing is driven by the potentially severe economic consequences due to unsecured embedded applications.

The following section is organized as follows. Section 2 discussed on physical design of the sensor node. Section 3 briefly discussed the identity based key agreement scheme. Section 4 introduces the proposed security framework followed validation of proposed protocol using AVISPA. Section 5 presents the significant benefits in memory utilization for the scheme and finally forming the conclusion in section 6.

2. Physical Design of the Sensor Node

According to TCG documents [3], trust is defined as an entity that always behave in the expected manner for the intended purpose while measurement is the process of obtaining the identity of the entity. From the TCG definitions, identity is equal to measurement. Therefore, for base station to trust the sensors that exist in its networks, the sensor nodes need to measure the entities in its platform, produce a measurement value and report the value to the base station. If the value reported is equal with the value measured before deployment, the base station will add the sensor node ID in the trust list thus enable the sensor node to participate in the networks.

In this process, the Root of Trust (ROT) which is an entity that must be trusted is located in the on-SoC ROM of the ARM11 processor. The integrity (I) of the image that is loaded into it which is the 1st Boot loader is assumed to be unmodifiable and therefore is always TRUE. The sensor node will only be able to complete the secure boot process only when the integrity in each level is true. The integrity of each level of operating system is evaluated by a set of equations. The final level secure boot equation takes into accounts the integrity of each layer below it, as shown in (1). Finally the value generated from the secure boot process will then be used to establish trust relationship with the base station. Details on the secure boot process can be found in [5].

$$I = 1 \text{ if } (I_0 \ \& \ I_1 \ \& \ I_2) = 1 \tag{1}$$

The main reason for having a secure boot process and secure memory devices is to protect the node from physical attack. While the secure boot process confirms the integrity of the images running in the nodes, the following section discuss communication procedure in confirming secured communication between sensors and base station.

3. ID-Based Key Agreement Scheme

ID based key agreement scheme is based on an Elliptic Curve Cryptography (ECC) type algorithm. IBE was proposed by Adi Shamir in 1984 and only in 2001, Boneh and Franklin [6] has successfully come out with a fully functional identity-based encryption scheme. IBE has simplified the certificate based public key encryption scheme by using publicly known unique identifiers to derive public keys and eliminate the needs of certificate authority.

In IBE, an arbitrary string is used as a public key. Public key can be calculated from any string such as email, project name or any other string. According to RFC 5408 [7], an IBE public key can be calculated by anyone who has the essential public while a cryptographic secret (master key) is needed to calculate an IBE private key, and the calculation can only be performed by a trusted server that has this secret. In WSN, the trusted authority or trusted entities is the base station which has to be placed in the most secure place and controlled directly by the network proprietor. Besides that, the existence of pre-deployment stage has offer better security and control environment for the key distribution phase. This criterion does not exist in other PKC infrastructure. Another characteristic that differentiates IBE from other server-based cryptography is that no communication is required with the server during encryption operation whereby the sender only needs to know the recipient’s ID for it to encrypt the message. Moreover, IBE implementation also consumes less memory for storing public keys of the other nodes.

4. Proposed IBE-Trust Security Framework

The following section will briefly discuss proposed identity based key distribution scheme to establish the security goals discussed earlier. The proposed scheme is motivated by trusted platform basic functions that measure and report the integrity of the platform which in this case is the base station [8]. With the nature of WSNs such as distributed, unsupervised and physically exposed, the need for a trusted platform is a must. By integrating trusted platform technique in the proposed security framework, base station can rely on the data it received from the sensor node.

Two basic trusted platform functions that we integrate into our security framework are: *Protected capabilities* that grant the user issuing the access command to protected locations such as in memory and register and *Integrity measurement and reporting*: Process of obtaining metrics of the platform characteristic (hash management value) and reporting to base station.

Simplified version of the protocol is best viewed in Fig. 1. Following section will discuss IBE_Trust protocol and verify the protocol using AVISPA tools. Noted that trusted is establish through two different processes which are secure boot and reporting the generated value to base station. Following sections present two cases that we consider in the development of the IBE_Trust protocol followed by the analysis of the protocol using AVISPA tools.

Case 1: Sensor node reports its trust value to base station.

Throughout this analysis we will consider that before deployment, a node public key and common parameters have been saved in each sensor node. Base station also have a list of sensor node’s ID and its HmVx value (value generated after secure boot process before deployment).We also adopt the only intruder model in AVISPA which is Dolev-Yao that is able to perform overhear, intercept, alter or inject any message into the radio communication channel.

Fig. 1 depicts the IBE_Trust model for the Case 1. This process happens immediately upon node deployment at the intended location. The successfully boot up node will report its measurement value (HmVx) to the trusted authority which in this case in the base station. Processes number 2 and 3 happened in the sensor node and base station will authenticate the sender with the newly HmVx value.

The protocol using AVISPA syntax is as follows:
A → BS : Snd(A.BS{Na’.Hm’}_Kbs)
A: Request BS to authenticate on Hm value
BS : Rcv(A.BS.{Na’.Hm’}_Kbs) in (A.Hm, trustmap) % BS compare the receive Hm value with the value in trustmap list.
BS→A: Snd({Na’.Nb’}_Ka)
Bs: Request A to authenticate on Nb Value
A: Rcv ({Na.Nb’}_Ka)
A → BS: Snd ({Nb’})_Ks
A: Request server to authenticate on Na value

As mentioned earlier, our protocol is based on IBE concept and therefore all the packet sent are encrypted with public key of the receiver which is the hash of the receiver ID. In this case, the sensor node, only need to store the ID of the BS. To confirm confidentiality of the HmVx value, the packet is encrypted with BS public key. Therefore only BS can decrypt the packet and compare the received value with the value in the list.

The above processes produce attack on the authentication of the HmVx value. Instead of BS send the start signal to A, the intruder has initiated the communication and asks A to send the HmVx value to intruder. Although the secrecy of HmVx value is not compromised, we have modified the protocol to prevent intruder from initiating the communication.

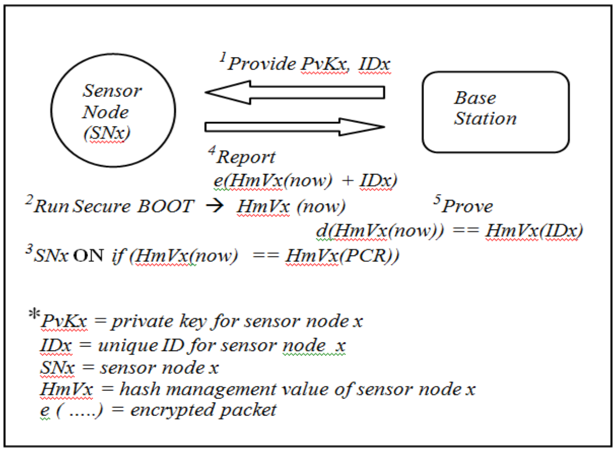
Sensor
Node
(SNx)
¹Provide PvKx, IDx
Base
Station
⁴Report
e(HmVx(now) + IDx)
²Run Secure BOOT → HmVx (now)
⁵Prove
d(HmVx(now)) == HmVx(IDx)
³SNx ON if (HmVx(now) == HmVx(PCR))
*PvKx = private key for sensor node x
IDx = unique ID for sensor node x
SNx = sensor node x
HmVx = hash management value of sensor node x
e (…..) = encrypted packet

Fig. 1: IBE-Trust Model

A → BS : Snd(A.BS.{Na'.Hm'}_Kbs)
BS : Rcv(A.BS.{Na'.Hm'}_Kbs) in (A.Hm, trustmap) % BS compare the receive Hm value with the value in trustmap list.
BS→A: Snd({Na'.Nb'}_Ka)
Bs: Request A to authenticate on Nb Value
A: Rcv ({Na.Nb'}_Ka)
A → BS: Snd ({Nb'})_Ks
A: Request server to authenticate on Hm value.

In the modified protocol, BS will only authenticate on the HmVx value once the authentication on Na and Nb are successful. This protocol reported no intruder attacks and the goal of the protocol which are authentication on HmVx and secrecy of HmVx value is accomplish.

Upon successful authentication, BS will generate a new list contain trusted node's identity. This new list, which is smaller than the trust list can be distributed to sensors in its network or stored in base station for faster verification process. Nodes will only communicate with the nodes which its ID exist in the new list. Trusted nodes in the networks remain to be in trusted condition as long as it remains in the ON state. Once reboot or shutdown due to any reason, the nodes will needs to re-authenticate with BS. Failure to authenticate will lead to node termination process where the node's ID will be removed from the trust list. All new nodes have to go through the same process before joining the group. Case 2 describes communication between sensor nodes.

Case 2: Communication with immediate neighboring node or Base Station

Two slightly different methods are discussed: The first method performs authentication by letting the sensor node keep a list of all trusted node IDs in its memory while in the second method receivers checks the authenticity of the sender with the BS. The first leads to extra memory requirement in sensor node and add extra security entities (item need to be protected) and the latter contributes to extra communication overhead.

The packet send by sender will include the source and destination address, its ID (ID_x, nonce value and encrypted data (encrypt with BS public key). Receiver then verifies the ID_x with the list of ID in his trustlist. If the ID is not valid, the packet will be discarded. In the later technique, instead of checking the trustlist in its memory, receiver will verify the validity of sender ID with the base station. BS will reply to receiver on the status and proceed to the next stage depending on the BS report. The nonce value is used to avoid replay attack in the network. Above protocol using AVISPA syntax is as follows:

(The HLPSL codes are simplified to reduce space).

1st technique:

```
A: Snd(A.B.IdA{Data.Na'}_Kb
B:Rcv(A.B.IdA{Data.Na'}_Kb/\ in (A.IdA,trustlist)
B:Snd(B.A.Ok)
A: proceed with next packet
```

2nd technique:

```
A: Snd(A.B.IdA{Data.Na'}_Kb
B:Rcv(A.B.IdA{Data.Na'}_Kb
   Snd(B.S.A.IdA)
BS:Rcv(Snd(B.S.A.IdA) /\ in (A.IdA,trustlist) =|>
     Snd(S.B.Ok)
B:Rcv(S.B.Ok) =|> Snd(B.A.Ok)
A: proceed with next packet
```

It is clear that the 1st technique consume less communication overhead compared to the 2nd techniques. Both techniques report no attacks on data secrecy and authentication. Further studies on energy and memory utilization are needed to confirm which technique performs better.

5. Reduced Node Memory Requirements

In this section, we evaluate the memory utilization for storing keys of our IBE-based key management scheme. The key pre-distribution scheme proposed by Eschenauer and Gligor[9], Du et.al[10, 11], and TinyIBE[12] is used for comparison. Although the WSN architecture presented above is between sensor node and base station only, our scheme is flexible to be implemented in hierarchal structure that may consist of base station, sensor node and cluster head.

Terminology used:

M = Number High end sensor (H) which is usually the cluster head

N = Number of Low end sensor (L)

In IBE-Trust scheme, each L and H sensors is preloaded with its private key, ID and equation to derive public key based on the ID. Thus H and L sensor is pre-loaded with a single key only. The total number of pre-loaded keys is:

$$M + N \tag{2}$$

In the E-G scheme, each sensor is pre-loaded with m keys. The total number of pre-loaded keys in a network with M+N sensors is:

$$m*(M+N) \tag{3}$$

In the Du et.al scheme (Asymmetric Pre-Distribution), total number of pre-loaded keys is: Where m = number of pre-loaded keys in sensors.

$$xM + yN \text{ where } xy=m^2 \tag{4}$$

Latest, Du et al. present a routing driven key-management scheme and the total number of pre-loaded keys is:

$$M(N+3) + 2N \tag{5}$$

Finally, the TinyIBE total number of keys is fixed at:

$$3M \tag{6}$$

From the above equations, the E-G and A-P scheme have m numbers of pre-loaded keys in each sensor node. Therefore, total numbers of keys in the networks increased linearly with the number of sensor nodes (H and L). This does not happen in IBE implementation where the numbers of keys mainly depends on the number of H and L sensors only. Fig.2 present the comparison between the describe schemes. The parameters were set as: M = 30, m = 100, x = 500, y = 20. The resulting graph shows that our IBE-Trust scheme gracefully scales with the number of nodes in the network. However, TinyIBE scheme produces a constant number of keys because no private key is stored in the sensor node. Instead they use a session key for communications with primary cluster head and they assume the cluster head to have more battery power compared to L sensors for generating numbers of session key for each L sensor.

We can also see that, the routing driven scheme (ECC) proposed by [11] increase almost the same as the A-P scheme and at L=1500, total number of keys in ECC exceed A-P scheme (not shown in graph). This suggests the need for proper key management or key distribution scheme in IBE implementation to avoid linear increase in total number of keys in the networks.

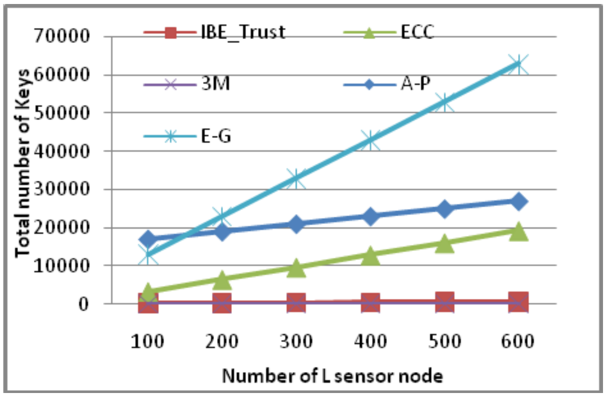

Fig.2: Pre-Loaded Keys in Networks

6. Conclusion

This work proposed a security framework for WSN applications that that need high security features. This might include applications such as in body sensor networks, oil and gas, crucial financial information, un-critical military communications, medical data, and others.This paper focuses on: *confirming the integrity of images on the platform through secure boot process,establishing trusted platform status by reporting the hash management value to base station*, and *finally presenting an analysis of the IBE-Trust framework using AVISPA tool to confirm message confidentiality and secure authentication between sensor node and base station and between sensors*. The trusted authentication mechanism proposed in the IBE_Trust framework will ensure the integrity of the data received in the data centric environment. Future work will be focused on real implementation of IBE_Trust framework and its performance analysis in real environment.

References

[1] W. Hu, P. Corke, W. C. Shih *et al.*, "SecFleck: A public key technology platform for wireless sensor networks," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Verlag, 2009, pp. 296-311.

[2] Y. M. Yussoff, and H. Hashim, “Trusted Wireless Sensor Node Platform,” in Proceedings of The World Congress on Engineering 2010 London, United Kingdom, 2010, pp. pp774-779.

[3] J. Grobschadl, T. Vejda, and D. Page, "Reassassing the TCG Specifications for Trusted Computing in Mobile Embedded Systems." pp. 84-90.

[4] T. Alves, D. Felton, and ARM, "TrustZone: Integrated Hardware and Software Security," *Technology in-Depth,* 3, 2004].

[5] H. Hashim, Y. M. Yussoff, and L. H. Adnan, "Secure Boot Process for Wireless Sensor Node."

[6] D. Boneh, and M. Franklin, “Idenetity-based encryption from weil pairing,” *Advance in cryptology-crypto,* vol. 2139, pp. 29, 2001.

[7] L. Martin, G. Appenzeller, and M. Schertler, "RFC5408 - Identity-Based Encryption Architecture and Supporting," Network working Group, 2009.

[8] D. Challener, K. Yoder, R. Catherman *et al., A practical Guide to Trusted Computing*: IBM Press, 2008.

[9] L. Eschenauer, and V. D. Gligor, “A key-management scheme for distributed sensor networks,” in Proceedings of the 9th ACM conference on Computer and communications security, Washington, DC, USA, 2002.

[10] X. Du, Y. Xiao, M. Guizani *et al.*, “An effective key management scheme for heterogeneous sensor networks,” *Ad Hoc Networks,* vol. 5, no. 1, pp. 24-34, 2007.

[11] D. Xiaojiang, X. Yang, C. Song *et al.*, "A Routing-Driven Key Management Scheme for Heterogeneous Sensor Networks." pp. 3407-3412.

[12] P. Szczechowiak, and M. Collier, "TinyIBE: Identity-based encryption for heterogeneous sensor networks." pp. 319-354.