2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks

# Design of A Key Establishment Protocol for Smart Home Energy Management System

Yue Li
School of Computer Science and Technology,
Donghua University,
No. 2999, North Renmin Road, Shanghai, China
E-mail: yueli1224@gmail.com

*Abstract*—**With the fast development of Wireless Sensor Networks (WSNs) and RFID technology, many Internet of Things (IOT) applications have been deployed in recent years. Smart home energy management systems form part of the smart grid program and are a fast developing smart home application area. Inadequate security is a big issue in smart home energy management systems. Most security protocols widely used for computer network and internet security cannot be implemented in smart home energy management systems as they are computational expensive for the wireless sensor nodes used in smart home applications. The major issue in the security of smart home energy management systems is the establishment of the initial session key between the wireless nodes and control center. In this paper, we propose a lightweight key establishment protocol for smart home energy management systems and present the implementation details of the protocol.**

***Keywords - wireless sensor network; smart home energy management system; key establishment***

## I. INTRODUCTION

Currently, wireless sensor network technology and Radio Frequency Identification (RFID)[1] technology are fast developing. Many objects equipped with wireless sensor nodes and RFID chips can communicate wirelessly with each other, gather information and react with command/request from the command center or controller. With the advantage of these advanced technologies, many creative and valuable applications can be designed and implemented, such as smart grid, mobile payments, smart health and smart traffic, etc. Smart home energy management (SHEM) system [2] is also a part of the smart grid program, which is being deployed worldwide.

Inadequate security is a big issue in the SHEM system. Most security protocols widely used in computer networks and internet cannot be implemented in the SHEM system as they are computational too expensive for use with wireless sensor nodes. The major issue with the security of the SHEM system is the establishment of the initial session key between the wireless nodes and control center. In this paper, we propose a lightweight group key establishment protocol for smart home applications and specify the implementation and evaluation of this protocol.

Security services such as authentication and key management are critical to communication security in the smart home network. In traditional networks such as the Internet, Public Key Cryptography (PKC) has been the important technology underlying many security services and protocols (e.g., SSL[3] and IPsec [3]). For example, PKC has been used to bootstrap symmetric session keys and authenticate messages to multiple receivers. However, in smart home network, PKC has not been widely adopted due to resource constraints on wireless sensor nodes.

As PKC is so important for access control, authentication and key establishment, it is desirable to explore the application of PKC on resource constrained sensor platforms.

As ECC (Elliptic Curve Cryptography) has proven to be applicable in resource constrained sensor nodes, many key establishment protocols based on ECC have been proposed [4, 5]. In 2004, Huang et al. [4] proposed an efficient Authenticated Key Exchange (AKE) protocol for WSNs which consists of many sensor nodes and security managers. The protocol is based on a combination of ECC and symmetric-key operations. This hybrid key exchange protocol reduces the high cost elliptic curve random point scalar multiplications at the sensor side and replaces them with low cost and efficient symmetric-key based operation. However, a security manager could learn the private key of a sensor after launching one normal run of the protocol with it [6].

There have been a few recent attempts to implement ECC-based security protocols in various wireless sensor platforms [7, 8, 9]. The results demonstrate that ECC is feasible on wireless sensor motes such as MICAz, TelosB and Tmote. Therefore, ECC cryptography will be adapted in our key establishment protocol.

The rest of the paper will be organized as follows. The network architecture of the smart home energy management system is illustrated in Section 2. The ECC-based key establishment protocol designed for the SHEM system is presented in Section 3. The implementation detail is described in Section 4. The summary of the paper is presented in Section 5.

## II. THE SMART HOME ENERGY MANAGEMENT SYSTEM

Consumer-side energy management is a part of the smart grid program. The customer can control the power consumption according to the power line's load, Smart-grid-enabled smart homes with time-of-use metering and energy-management devices and tools help consumers monitor, manage and control energy usage, while helping them optimize performance and reduce energy losses from major appliances, heating, cooling and lighting. With a smart home energy management system, consumers can manage energy usage and costs throughout the day, without compromising their lifestyles.

The network architecture of SHEM system is illustrated in Figure 1.

978-0-7695-5042-8/13 $26.00 © 2013 IEEE
DOI 10.1109/CICSYN.2013.42

IEEE computer society

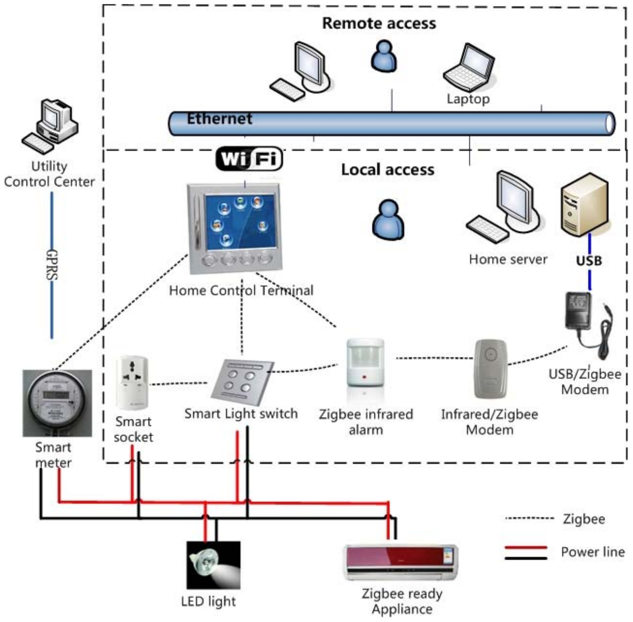


Figure 1. Network architecture of a smart home energy management system

This system consists of the following devices:

- A smart meter is the core device in this home energy management system. Comparing to a normal power meter, it can provide more detail of power consumption of consumers. A smart meter is not only used for automatic meter reading, but also involves many other technologies and functions, such as real-time sensor technology, low-power wireless communication technology, power quality monitoring, utility information broadcasting terminal etc.
- A smart socket can be mounted in the wall as a normal electricity socket. Smart sockets are able to measure the power consumption of the appliance and communicate with the neighboring Zigbee devices.
- A smart light switch is Zigbee enabled and used for remote and local access through the home control terminal, local server and web page. It also supports multiple-scene settings using scene buttons.
- An Infrared/Zigbee modem is used to convert the Zigbee commands to infrared signal. This device will help the local home network forward the commands to electric appliances like TV, air-conditions and Multimedia players.
- A USB/Zigbee modem is used to forward commands from the local server and web pages to the local Zigbee network.
- A Zigbee based infrared alarm is used for door and window access monitoring. The alarm signal will be sent to the local server and control terminal.

Smart meter, smart socket, smart light switch, USB/Zigbee modem, Infrared/Zigbee modem and Zigbee infrared alarm are all based on MSP430 microcontrollers and CC2430 receivers. These devices are typical of wireless sensor nodes, which are small in size and with constrained computational resource. This home network can be considered as a small Wireless Sensor Network (WSN), in which each message can reach its destination node by multi-hops. The conventional PKC key establishment such as the key establishment protocols in SSL and TLS cannot directly be implemented on the Zigbee-based local home network. A new ECC-based group key establishment protocol is proposed in Section 3.

## III. THE ECC-BASED KEY ESTABLISHMENT PROTOCOL FOR A SHEM SYTEM

The Home Control Terminal (HCT) is based on the ARM architecture and powered by a 2600mAh battery with DC in charging socket, which means that the HCT has more computational resources than the MSP430 based devices. We label all the MSP430 based Zigbee devices as nodes. We also make HCT in Figure 1 act as a security manager in the home network. The security manager is granted special capabilities to assist in provisioning keys to other devices. It should establish an initial link key with the Zigbee node before it can install link keys into that node for secure communication with other devices.

The ECC-based Authenticated Key Establishment (EAKE) protocol presented in this section is based on the protocol presented in [4]. With some modifications and improvements, our proposed EAKE protocol provides mutual authentication and key freshness.

There are two phases in the EAKE protocol. In the first phase, each party (e.g. node or a security manager) in the system securely communicates with the Certificate Agent (CA) through an out-of-band channel and gets private/public key pair. In the second phase, a node and a security manager carry out an authenticated key exchange protocol for establishing an initial session key.

### A. Notation used in the protocol

***P*****:** denotes a base point of large order $n$ selected for elliptic curve, which is public to all users.
***U*****:** denotes the identity of node $U$,
***V*****:** denotes the identity of security manager $V$,
***H*****( ):** denotes a secure one-way hash function which maps finite binary strings to integers in the range [2,n-2].
**KDF():** denotes secure key derivation function,
**MAC():** denotes a message authentication code function
||: denotes the conventional binary string concatenation operator.
**E(*M, K*):**denote the symmetric encryption of Message $M$ using key $K$.
**($q_U$, $Q_U$):** denote the private/public pair of node $U$, where $q_U$ is a random integer and $Q_U = q_U \times P$, also $q_U = Q_U^{-1}$
**($q_V$,$Q_v$):** denote the private/public pair of security manager($V$), where $q_V$ is a random integer and $Q_V = q_V \times P$, also $q_V = Q_V^{-1}$ .
**$IC_U$:** denotes the implicit certificate of node $U$.

***$IC_V$*:** denotes the implicit certificate of security manager *V*.

### *B. Key generation*

In the key generation phase, the security manager gets its long-term public key pair($q_V$,$Q_V$) through the out-of-band channel from the Key Distribution Center (KDC). In addition, the security manager also gets the public keys of client nodes from the KDC. In the same way, each client node gets its long-term public key pair ($q_U$,$Q_U$) and the security manager's public key through the out-of-band channel.

### *C. The EAKE Protocol for Initial Session Key*

After having public key pairs generated, sensor *U* and security manager *V* carries out the following protocol for establishing an initial session key. The session key will be used to setup a secure channel for *V* to install link keys or group key to *U*.

(1) Sensor *U* randomly picks a *k*-bit integer $K_U$ and a $(160-k)$ bit integer *r*, computes $d_U \leftarrow H(K_U \| r)$ , $D_U$ = $d_UP$, $R$ = $d_UQ_V$ , and sends $D_U$ and $T=(K_U \| r)\oplus R.x$ to $V$ , where *R.x* denotes the *x* coordinate of *R*, and sends the encrypted message *T* to *V* with a signature of *U*, $Sig_U\{T\}$.The protocol adopts Shamir and Tauman's online/offline [10] signature scheme to reduce the computational complexity of the lower-power nodes.

(2) Security manager *V* computes $R=q_V D_U$ , $K_U \| r = T \oplus R.x$ and authenticate the identity of *U* using the signature. Then, security manager checks if $D_U = H(K_U \| r)P$ . If yes, *V* obtains $c_U$ as the most significant *k* bits of m; otherwise, the protocol is terminated. When the decryption and authentication completes, security manager *V* generates a *k*-bit random number $C_V$ and sends encrypted message $E((V, C_V, r), K_U)$ to *U*, where $E((V, C_V, r), K_U)$ is a secure symmetric key encryption function under the key $K_U$. *V* also computes $KDF(K_U \| c_V \| U \| V) \rightarrow K_{UV} \| sessionK$ , where $K_{UV}$ is the MAC key for *U* and *V*.

(3) Sensor *U* decrypts the message received from *V* and gets value $C_V$. *U* then computes $KDF(K_U \| c_V \| U \| V) \rightarrow K_{UV} \| sessionK$ and sends $MAC(U \| V \| r, K_{UV})$ to *V*.

(4) Security manager *V* checks if the MAC is valid, if yes, *SessionK* is established as the initial session key between *U* and *V* . *V* sends a encrypted message $E(U \| V, SessionK)$ under *SessionK* to *U* in order to inform *U* that the session key *SessionK* is established.

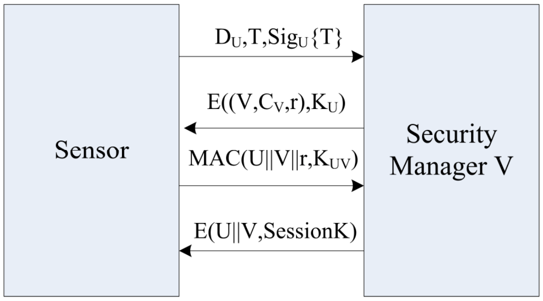


Figure 2. The EAKE protocol for initial session key

### *D. Security evaluation and Comparisons*

The security manager authenticates the sensor *U* at the start of the protocol. This prevents the protocol from an impersonation attack in the early stages of the protocol. The EAKE protocol also avoids using equation $z = q_U H(K_{UV}^{'}) + d_U (\bmod n)$ at step 3 in the protocol presented in [4], which makes the private key of node *U* revealed by security manager *V*. Instead, $MAC(U \| V \| r, K_{UV}^{'})$ is applied, as both U and V has knowledge of *MAC key*, $K_{UV}^{'}$ , at that stage, if *V* can compute the same MAC value of *U*||*V*||*r* under MAC key $K_{UV}^{'}$ , which means the verification of message*(U*||*V*||*r)* holds; this confirms that both *U* and *V* agree on the same session key.

Recent related key establishment protocol for wireless sensor networks are presented below. Comparisons between our protocol and other protocols are summarized in Table 1.

In 2011, Eldefrawy et al. proposed a key agreement algorithm with rekeying for WSNs using ECC and RSA public key cryptography [11], this protocol is also dependent on a specific routing protocol. As a result, the protocol only establishes a pair-wise key between nodes in a specific route, which avoids establishing a pair-wise key for each of the neighbours. The shortcomings of Eldefrawy et al's protocol are that the protocol has to be used with a specific routing protocol and the higher communication overhead due to the use of both RSA and ECC in key establishment.

Using the concept of Schnorr Signature [12] and based on ECC, Huang et al. in [13] designed a key establishment in the authentication procedure of the access control scheme for WSNs. The new designed key establishment in [13] also used the concept of timebound in which once a time period has elapsed, the sensor node in the wireless sensor network cannot access any data for a future time period in order to protect future messages. Nevertheless, adversaries can still apply sensor node replication attacks in the period of expiration time $w_i$. The reason being that the adversary can compromise the sensor node and apply a replication attack before expiration time $w_i$ is reached.

Hang et al. in [14] designed a new identity-based DH key agreement protocol for wireless sensor networks Based on the Arazi-Qi Scheme, Authors prove that this key establishment scheme is resilient against the masquerade attack, ephemeral key compromise attack and provide perfect forward secrecy, but the protocol is not proved to be resilient against replication attack. Besides, this protocol requires high computation resource on sensor nodes according to the performance analysis. Comparisons between our protocol and other protocols are shown in Table 1.

TABLE I. SECURITY COMPARISON AMONG THE REFERENCED PROTOCOLS

| Item | Proposed | Eldefrawy et al.'s | Hang et al.'s | Huang et al.'s |
|---|---|---|---|---|
| Group Key agreement | Yes | No | No | No |
| Mutual authentication | Yes | Yes | Yes | Yes |
| Avoiding replication attack | Yes | Yes | No | No |
| Avoiding masquerade attack | Yes | Yes | Yes | Yes |
| Forward secrecy | Yes | No | Yes | Yes |

.

## IV. IMPLEMENTATION AND EVALUATION

The EAKE protocol is implemented and evaluated on the trial smart home energy management system.

The experiment network contains seven client nodes (smart meter, smart sockets, Zigbee infrared alarm, Zigbee/infrared modem, Zigbee/USB modem) and one security manager (HCT). Using the HCT as the security manager enables the security manager to implement all operations by the Java program and store all members' public keys in the local memory device without worrying about memory constraints. This method reduces the execution time of the protocol and releases the memory and power constraints existing in smaller nodes.

The protocol is written in C on the nodes and programmed in Java on the security manager. To implement the MAC, block cipher, ECC point addition and scalar multiplication in the implementation of key establishment protocols.

The flow chart of the proposed key agreement protocol implementation is illustrated in Figure 3. The implementation is divided into two modules, the client (group member) module and the security manager module.

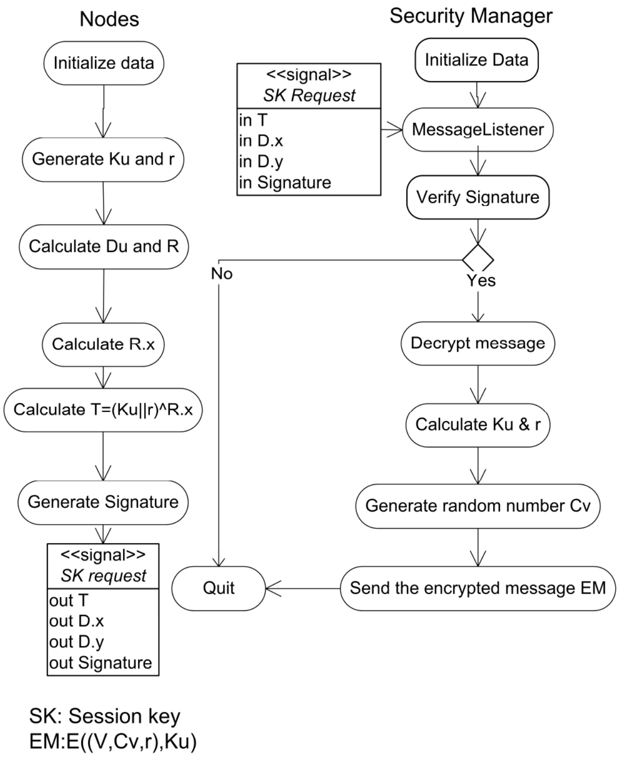
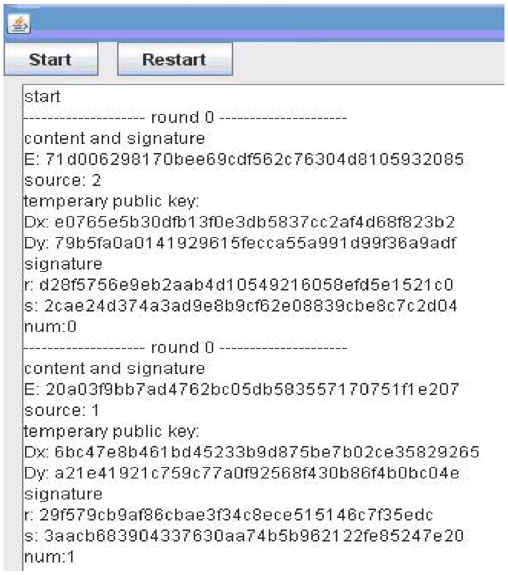


Figure 3. The flow chart of the EAKE protocol implementation

- The client module implements all the operations required by the proposed protocol on the client side, which involves ECC point multiplication, ECDSA signature generation and MAC generation.
- The security manager module has two parts. The first part is written in C and implemented on Zigbee/USB modem, and the other part is securitymanger.java which is written in Java and implemented on the security manager . These two parts are linked by a Java class MoteIF which enables Java applications to send and receive the message through UART (Universal Asynchronous Receiver/Transmitter).

We developed a control and display GUI panel for monitoring the handshake messages during the EAKE protocol. A snapshot of the GUI panel is displayed in Figure 4. It can be seen that the Control/display panel displays the encrypted message *E*, the ephemeral shared point *R*, and the ECDSA signature generated by the sensor nodes.



```
start
-------------------- round 0 ---------------------
content and signature
E: 71d006298170bee69cdf562c76304d8105932085
source: 2
temperary public key:
Dx: e0765e5b30dfb13f0e3db5837cc2af4d68f823b2
Dy: 79b5fa0a0141929615fecca55a991d99f36a9adf
signature
r: d28f5756e9eb2aab4d10549216058efd5e1521c0
s: 2cae24d374a3ad9e8b9cf62e08839cbe8c7c2d04
num:0
-------------------- round 0 ---------------------
content and signature
E: 20a03f9bb7ad4762bc05db583557170751f1e207
source: 1
temperary public key:
Dx: 6bc47e8b461bd45233b9d875be7b02ce35829265
Dy: a21e41921c759c77a0f92568f430b86f4b0bc04e
signature
r: 29f579cb9af86cbae3f34c8ece515146c7f35edc
s: 3aacb683904337630aa74b5b962122fe85247e20
num:1
```

Figure 4. Java control/display panel (GUI application) of the protocol implementation

Time for each link key establishment and power consumption are taken into account in the course of the evaluation of the EAKE protocol. In order to measure the energy consumption for each experiment, the Agilent 66321D Mobile Communication DC Source with battery emulation is employed. The Agilent 66321D is designed for testing next generation wireless communications products, and can be used in place of a battery. The example of energy consumption measurement using the Agilent 66321D is displayed in Figure 5.

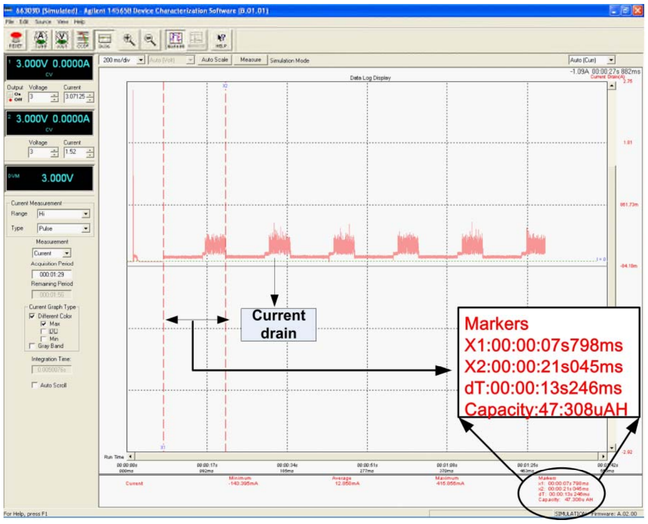


Figure 5. Measuring energy consumption using the Agilent 14565B device characterization software

In the evaluation, The 128-bit and 160-bit ECC parameters recommended by SECG [15] are chosen for use in the experiments presented in this section, while the 192-bit ECC parameters are not included in the evaluation. This is because the 192-bit ECC requires 48 bytes to represent the point (public key pair) on the curve, which results in 120 bytes payload in the communication message, this will exceed the maximum payload size of the Zigbee node. Both time delay and power consumption of the Zigbee nodes during the initial link key establishment are evaluated. The evaluation results are listed in Table 1.

TABLE II. THE EVALUATION RESULTS OF THE EAKE PROTOCOL

| **Elliptic curve** | ***Metric*** | |
|---|---|---|
| | ***Time(seconds)*** | ***Power consumption(μWH)*** |
| Secp128r1 | 7.52 | 113.03 |
| Secp128r2 | 7.64 | 113.66 |
| Secp160k1 | 8.54 | 134.58 |
| Secp160r1 | 8.22 | 131.37 |

## V. SUMMARY

A lightweight key establishment protocol for the smart home energy management system is proposed in this paper. To ease the monitoring authentication process during the protocol, a Zigbee packet monitor and control panel is developed to record the messages transmitted by the Zigbee devices. The implementation and evaluation results of the key establishment protocol are presented. The results demonstrate that the protocol is feasible for use in a smart home energy management system.

## ACKNOWLEDGMENT

This research is supported by the Fundamental Research Funds for the Central Universities

## REFERENCES

[1] R.Weinstein, “RFID: a technical overview and its application to the enterprise”, IT Professional,vol. 7, pp.27-33,2005.

[2] D.Han and J.Lim, “Smart home energy management system using IEEE 802.15.4 and Zigbee”, Consumer Electronics, IEEE Transactions on Comsumer Electronics, vol.56,pp.1403-1410,2010.

[3] G. Xu, M.Zhang and J.Peng, Network Security, $2^{nd}$ ed., 2007,pp.182-197.

[4] Q. Huang, J. Cukier, H. Kobayashi, B. Liu and J. Zhang, “Fast authenticated key establishment protocols for self-organizing sensor networks,” the Second ACM International Conference on Wireless Sensor Networks and Applications. 2003.

[5] H.Yeh,T.Chen,P.Liu,T.Kim and H.Wei, “A Secured Authentication Protocol for Wireless Sensor Networks Using Elliptic Curves Cryptography”,Sensors 2011, vol.5, pp.767-779, 2011

[6] Y. Li and T. Newe, “Key Exchange Protocol for Wireless Sensor Network: Formal Verification using CSN Modal Logic”, IEEE Sensors Applications Symposium (SAS 2008), Atlanta, Georgia, USA, pp.193-198. 2008

[7] M.Aydos, T.Yan and C. K. Koc, “A High-speed ECC-based Wireless Authentication Protocol on an ARM Microprocessor”, 16th Annual Computer Security Applications Conference (ACSAC'00), New Orleans, Louisiana,2000.

[8] R. Struik and G. Rasor, “Mandatory ECC Security Algorithm Suite,” IEEE P802.15 Wireless Personal Area Networks, 2002.

[9] A. Liu and P. Ning, "TinyECC:A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks", 7th International Conference on Information Processing in Sensor Networks (IPSN 2008),pp.245-256, 2008.

[10] A. Shamir and Y. Tauman, "Improved on-line/off-line signature schemes", Advances in Cryptology— Crypto'01, Springer-Verlag, Berlin, pp.355-367, 2001.

[11] M. H. Eldefrawy, M. K. Khan, and K. Alghathbar, A Key Agreement Algorithm with Rekeying for Wireless Sensor Networks using Public Key Cryptography. In 2010 International Conference on Anti-Counterfeiting Security and Identification in Communication (ASID), Chengdu, 2010 (pp. 1-6): IEEE. doi: 10.1109/ICASID.2010.5551480.

[12] C. P. Schnorr, Efficient identification and signatures for smart cards. In Advances in Cryptology (Vol. 435, pp. 339-351): Springer, 1990.

[13] H. Huang, A New Design of Access Control in Wireless Sensor Networks. International Journal of Distributed Sensor Networks, 2011, doi:10.1155/2011/412146.

[14] I. Hang, M. Ullmann, and C. Wieschebrink, A new Identity-based DH Key-agreement Protocol for Wireless Sensor Networks Based on the Arazi-Qi Scheme. In WiSec'11, Hamburg, Germany, 2011: ACM

[15] Standards for Efficient Cryptography Group (SECG), "SEC 1: Elliptic Curve Cryptography", Version 2.0, May 2009