

# Wireless Sensor Network Protocols for Secure and Energy-Efficient Data Transmission

Sang-Eon Lee<sup>1</sup>, Sang-Ho Shin<sup>2</sup>, Geum-Dal Park<sup>3</sup>, Kee-Young Yoo<sup>2\*</sup>

<sup>1</sup>Dept. of Information Security, Kyungpook National University

<sup>2</sup>Dept. of Computer Engineering, Kyungpook National University

<sup>3</sup>Dept. of Electrical Engineering and Computer Science, Kyungpook National University  
{s4ng30n, shshin80, machunru2}@infosec.knu.ac.kr, yook@knu.ac.kr

## Abstract

*Sensor networks are expected to be used at anywhere in the near future and recently their security problems have been rising. Due to the limited computing power of sensor nodes, it is impossible to use asymmetric cryptography approaches for the session key establishment or to apply advanced encryption method such as AES and DES for the data encryption. In this paper, therefore, we propose a key establishment and a data encryption scheme for secure and energy-efficient data transmission in wireless sensor networks (WSNs). The proposed schemes just treat simple operations as a message authentication code (MAC), exclusive-OR (XOR) and time-spacing key derivation function (TSDF). Additionally, the security of the proposed protocols is analyzed.*

## 1. Introduction

Wireless sensor networks (WSNs) have currently been used for a variety of applications such as environment monitoring, health monitoring, military applications, etc. [1] and also WSNs are expected to be used at anywhere in the near future. To support many kinds of applications based on sensor networks, the consideration of security aspects is essential.

For the satisfaction of security aspects, it is better to apply well-known asymmetric cryptography such as RSA and Diffie-Hellman algorithms to key establishment and advanced symmetric cryptography such as AES or DES to data encryption. However, sensor nodes have low computing power, storage, bandwidth and energy [2]. SmartDust [3], for instance, consist of a 8-bit, 4MHz CPU with only 512 bytes RAM space for data, 8K instruction flash, and

4500 bytes of available code space. Because of their inherited resource limitation, advanced cryptographic methods cannot be applied to sensor nodes.

Various key distribution researches [2, 4, 5, 6, 7] and encryption of messages researches [3, 8, 9] were proposed for the satisfaction of security and energy-efficiency in WSNs. First, considering key distribution researches, key distribution schemes are classified into post-key and pre-key distribution according to the distributed time. In post-key distribution schemes, each shared key is established after each sensor node is deployed on the fields. Their key distributions are based on asymmetric cryptography approaches which need much more energy consumption and computing power [6, 7]. Although satisfaction of security is high, it is too heavy to apply to sensor node.

The pre-key distribution scheme is also classified into the use of one master key in all nodes or individual key between two parties. If every node use one master key, it is very efficient, but its security depends only on master key. So, the reveal of the master key brings a high degree of danger to all nodes. The method of using individual key between two parties is more secure than using one master key, but it has a key management problem in each sensor node. For instance, when a new node joins in a sensor network, its neighborhood also updates a new node's individual key [10].

A key negotiation scheme in SPINS [3] was proposed by A. Perrig et al. and BROSKE [11] was proposed by C. Lai et al. that they belong to a pre-key distribution scheme but after deployment, they establish a session key between two parties. In SPINS protocol, for the establishment of a session key between two parties, a trustful base station (BS) creates a session key and it is distributed to the two sensor nodes. Every session key is created and distributed through the base station. The traffic on sensor nodes near the base station is increased and as a result, the battery is rapidly consumed. In BROSKE, the sensor node who wants to estab-

\*Corresponding author: Kee-Young Yoo (Tel.: +82-53-950-5553; Fax: +82-53-957-4846)

lish a session key broadcasts a negotiation message which is encrypted by the master key to his one-hop-neighbors. Because of this broadcasting method, it has the efficiency of communication. However, when sensor nodes establish session key, this method totally depends on the master key and it cannot authenticate each other at every key establishment due to using same master key, i.e. their ID is open in WSNs and they use same master key. So they cannot be distinguished from each other.

Second, considering encryption researches, Hasan et al. [9] evaluated several cryptographic encryption algorithms such as AES, TEA (Tiny encryption algorithm), DES and Blowfish (a mini-version of Blowfish). Their studies reveal that AES and DES require a lot of memory space for lookup tables and that these encryption algorithms are beyond a sensor node's capacity. In SPINS, a counter mode is used to encrypt the message in which the compacted RC5 is coded. RC5 requires many memories on its key expansion steps and uses extreme circular shifts [12].

In this paper, therefore, it is proposed that a key establishment and a data encryption scheme for secure and energy-efficient data transmission in WSNs be used. The key establishment scheme is an advanced hybrid key establishment scheme which has not only the efficiency of using a unique secret key but also the security of using a random key of each pair nodes. The data encryption scheme which is suitable for WSNs uses only exclusive-OR (*XOR*) operation, message authentication code (*MAC*), and time-spacing key derivation function (*TSDF*) for efficiency and security.

This paper is organized as follows: In section 2, the security requirements are described. In section 3, the assumptions and notations are explained. In section 4, the proposed protocol is presented. Then, the proposed protocol is analyzed how efficient it can be and whether it can meet several security requirements or not in Section 5. Finally, the paper is concluded.

## 2. Security Requirements

In this section, the security requirements on sensor networks are described.

- **Confidentiality** Confidentiality means that some secret information should be protected against any third parties who are not certified. In the case of key establishments, they exchange secret information among nodes. This information should be encrypted, and only the nodes which have the shared key should be able to check the information. In this way, the confidentiality is satisfied.
- **Authentication** Authentication confirms the participant's identity, so it distinguishes the legal users from any potential attackers in the cyber communication

network. In the case of sensor networks, the data which was given and taken among each sensor node need verification whether the data come from a trustful sender. If not, it permits false data, and it causes trouble in the behavior of the network. Therefore, mutual authentication can protect this problem.

- **Integrity** Let the information be safe without any unexpected change in insecure networks by protecting the information. In order to guarantee the integrity, the sensor should become aware of any quick change of the data such as the insertion, deletion, and substitution by unauthorized third parties. Integrity should be guaranteed in many fields of sensor application such as pollution and healthcare monitoring because they require the exact result.
- **Freshness** It guarantees the freshness of the message. It means that the message should follow the order of the message, and do not reuse the information. For the freshness, network protocol should redesign the method to identify the duplicate packets, and to cast the message in order to prevent any possible mix-up. An attacker's insecure information gathering can disturb the freshness.

## 3. Assumptions and Notations

In this section, some essential assumptions and notations are mentioned. First, system assumptions and network restrictions are mentioned. It is assumed that

- WSNs are open networks, i.e. adversaries can eavesdrop on messages in communications.
- Each wireless sensor node has an opened ID and identical master key which is saved before it is deployed.
- The message of data transmission has the same length and its length is not too long. For example, the data size of TinyOS is 28 bytes as a default [13]. In this paper, a 32-byte key length and data length are used.
- The sensor nodes transmit information to BS when some events have occurred.

Next, the notations used throughout this paper as follows (Table 1) are defined.

## 4. The Proposed Protocols

In this section, we propose key establishment scheme and a data encryption scheme for secure and energy-efficient data transmission in WSNs.

**Table 1. Notations**

Notation	Meaning
$Node_i$	Sensor node $i$
$Node_*$	Each node of one-hop-neighbors
$ID_i$	Identity assigned to $Node_i$
$K$	Master key
$SK_{AB}$	Session key between $Node_A$ and $Node_B$
$M$	Transmission message
$\oplus$	An exclusive-OR operation
$\parallel$	A concatenation operation
$N_i$	Random number generated by $Node_i$
$T_i$	Timestamp generated by $Node_i$
$MAC_K(M)$	A message authentication code of message $M$ created by key $K$
$TSDF(S^{(t)})$	Time-spacing key derivation function ( $TSDF$ ) for making derivation key of $(t+1)$ th encryption, $S^{(t)}$ is $t$ -th derivation key.
$ESF(K)$	Extract seed function ( $ESF$ ) which is explained in subsection 4.2
$RCS^{(n)}(x)$	$n$ -bit right circular shift of $x$ , $RCS^{(n)}(x) = (x \gg n) \vee (x \ll 256 - n)$ , where $\vee$ is bitwise OR

#### 4.1. Key establishment scheme

The proposed key establishment scheme consists of two phases: an initial key establishment phase and a key update phase.

##### [Initial key establishment phase]

The initial key establishment phase of proposed key establishment scheme is illustrated in Figure 1.

1. When  $Node_A$  establishes an initial session key with his one-hop-neighbors,  $Node_A$  first chooses a random number  $N_A$ , and then computes  $V = MAC_K(ID_A \parallel N_A \oplus K)$ . After that, he broadcasts  $ID_A \parallel V \parallel N_A \oplus K$  to his one-hop-neighbors.
2. The receivers,  $Node_*$ , verify  $V \stackrel{?}{=} MAC_K(ID_A \parallel N_A \oplus K)$ . If the verification is right, they also choose random number  $N_*$  and compute  $X = MAC_K(ID_* \parallel N_A \oplus N_*)$ , and then reply  $ID_* \parallel X \parallel N_* \oplus K$ . Otherwise, the receivers reject the message and send request message again.
3.  $Node_A$  also check  $X \stackrel{?}{=} MAC_K(ID_* \parallel N_A \oplus N_*)$  and if the check is wrong, reject the message and send the request message again. Finally,  $Node_A$  and  $Node_*$

have an initial session key  $SK_{A*} = K \oplus N_A \oplus N_*$ , respectively.

##### [Key update phase]

In the key update phase, a new session key is established by the previous session key not the master key. In other words, each node sends a random number and  $MAC$  using the previous session key to the other node. The key update phase of the proposed key establishment scheme is similar to initial key establishment phase. So, the key update phase is simply described as follows:

$$Node_A \longrightarrow Node_B : ID_A \parallel V \parallel N_A \oplus SK_{AB}$$

$$SK'_{AB} = SK_{AB} \oplus N_A$$

1. When  $Node_A$  wants to a update session key with  $Node_B$ ,  $Node_A$  chooses a random number  $N_A$  and computes  $V = MAC_{SK_{AB}}(ID_A \parallel N_A \oplus SK_{AB})$ . After that, he gives  $ID_A \parallel V \parallel N_A \oplus SK_{AB}$  to  $Node_B$ .
2. The receivers,  $Node_B$ , verify  $V \stackrel{?}{=} MAC_{SK_{AB}}(ID_A \parallel N_A \oplus SK_{AB})$ . If their verification is right,  $Node_A$  and  $Node_B$  update session key  $SK'_{AB} = SK_{AB} \oplus N_A$  respectively. Otherwise, the receivers reject the message and send the request message again.

According to a security satisfaction of environment, the update of a session key happens periodically.

#### 4.2. Data encryption scheme

In order to explain the data encryption scheme, the relevant symbols are defined as follows.

**Definition 1**  $\forall n > 0$ ,  $\bigcup_{i=1}^n B^{(i)} = B^{(1)} \parallel B^{(2)} \parallel \dots \parallel B^{(n)}$ , for all  $n \in \mathbb{Z}$ ,  $B^{(i)}$  is 4 bits block.

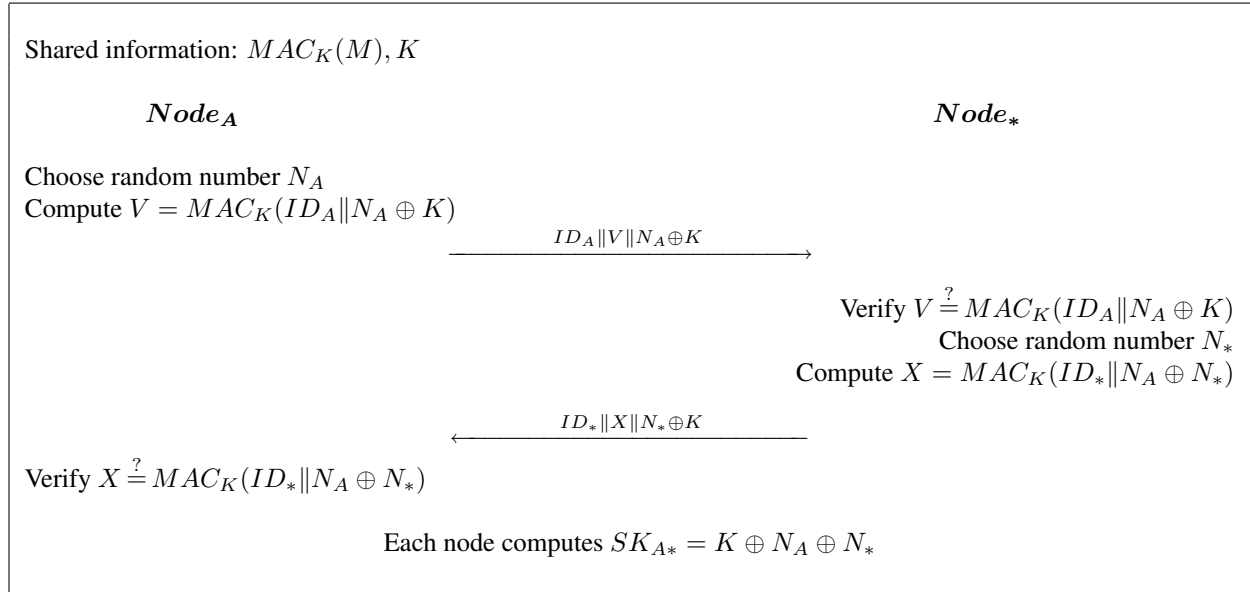
**Fact 1**  $S^{(t)} = \bigcup_{i=1}^{64} B^{(i)} = B^{(1)} \parallel B^{(2)} \parallel \dots \parallel B^{(64)}$

When  $Node_A$  transmits secret information to  $Node_B$ , the delivery message is as follows:

$$Node_A \longrightarrow Node_B : T \parallel D \parallel MAC_{SK_{AB}}(D \parallel T)$$

Where  $D$  is  $RCS^{(seed)}(TSDF(S^{(t)})) \oplus M$  (as shown Figure 2).  $Node_A$  makes  $MAC_{SK_{AB}}(D \parallel T_A)$  for data integrity and freshness, and sends to  $Node_B$  with a timestamp and encrypted message  $D$ . The numerical expression of  $TSDF$  is as follows:

$$TSDF(S^{(t)}) = \bigcup_{i=1}^{64} B^{((seed+1)i - seed \pmod{64})}$$



**Figure 1. Initial key establishment phase of proposed scheme**

Where  $S^{(0)} = SK_{AB}$  and the *seed* is extracted by *ESF* which uses a bit array of a session key and a 5-bit sliding window. The method of extracting *seed* is that binary value of 5 bits sliding window converts decimal value. After extracting *seed*, the sliding window shifts 5 bits and if it reaches the last bit of the bit array, it turns back to the first bit of the bit array, i.e. it makes a cycle. The number of shiftings serve as a counter. If the synchronization of data transmission is broken by errors of communication, the proposed counter exchange protocol in SPINS [3] is used. The *ESF* and *TSDF* is illustrated in Figure 3.

## 5. Analysis

In this section, the security of the proposed protocols are analyzed according to the requirements defined in the second section and the efficiency of proposed protocols.

### 5.1. Security analysis

In the key establishment scheme, *Node<sub>B</sub>* can get  $N_A$  from the  $N_A \oplus K$ , but attackers cannot get the value  $N_A$  because they do not have the master key  $K$  or the session key in the key update phase. In the data encryption scheme, each communicating node sends a message which is encrypted by the derivation key. Hence, a receiver node which has the same derivation key can decrypt the message exactly, but any adversaries cannot have the original message despite being able to intercept the message.

By using the *MAC*, the proposed protocols provide the authenticity. In the proposed protocols, *MAC* includes

node's *ID*. Each node checks whether its partner is valid or not using *MAC*. Even though the *ID* is opened, the adversary does not know the session key between two nodes, and he cannot get the correct value of the *MAC*. When the adversary sends the wrong value of *MAC*, the session will have failed.

The proposed protocols provide the message integrity by *MAC*. When *Node<sub>B</sub>* responds to the message for key establishment, *Node<sub>A</sub>* can decide the integrity of the message through  $N_B$  in *MAC*. Additionally, when data is transmitted,  $D$  with  $D$  in  $MAC_{SK_{AB}}(D || T_A)$  is compared and its integrity is verified.

The proposed protocols provide freshness by the change of session key and timestamp. In the key establishment scheme, even though adversaries try to reuse previous messages, the session key used in *MAC* is changed continuously. Therefore, the freshness of the key establishment scheme is satisfied. For example, if an adversary tries to use  $ID_A || MAC_K(ID_A || N_A \oplus K) || N_A \oplus K$  in another key establishment scheme, its key establishment scheme is terminated due to the updated session key. In the data encryption scheme, the derivation key is also changed continuously, but according to the sensor environment, the derivation key can be used repeatedly from the sliding window rotation. Hence, for complement of its problem, timestamp is used in the data encryption scheme. Through the timestamp, even if the derivation key is used again, the freshness of data encryption scheme is satisfied.

Finally, the encryption using *XOR* operation and the security of derivation key are analyzed. The *XOR* operation has the following property:

$$M \oplus K \oplus K = M$$

In other words, using the same two keys in the *XOR* operation makes an encrypted message be decrypted as below. The weak point of this method is well known [7].

$$E(A) \oplus E(B) = A \oplus K \oplus B \oplus K = A \oplus B$$

Although adversaries get a lot of collection like  $A \oplus B$ , they can find some patterns and get some secret information partially but not exactly. But, if different keys are used, the message is not revealed. Therefore, the proposed key encryption scheme provides good security due to using the derived key from the key spacing function.

Additionally, brute force attack which is a method of defeating a cryptographic scheme by trying a large number of possibilities can be considered. In Wi-Fi Protected Access, because its security depends on key size, to protect against a brute force attack, a random key of at least 20 bytes should be used, and 33 bytes or more is recommended [14]. Our protocols can protect against a brute force attack because 32 bytes of key size were used and a session key is updated by periods.

## 5.2. Efficiency analysis

It is important to minimize the computational cost of a sensor node. In the initial key establishment phase, the broadcasting method is used. If one node has  $n$  one-hop-neighbors, it is  $\frac{1}{n} \times 100$  (%) more energy efficient than the normal method approximately. The computational cost and the communication cost of key establishment scheme are described in Table 2.

In the data encryption scheme, the sensor node of our scheme requires just a 256-bit *XOR* operation, one rearrangement of the session key, and a right circular shift. It

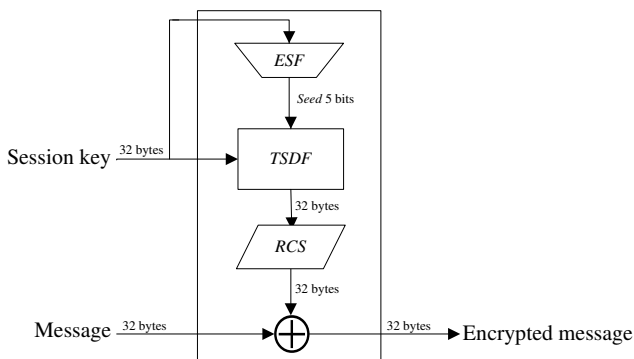
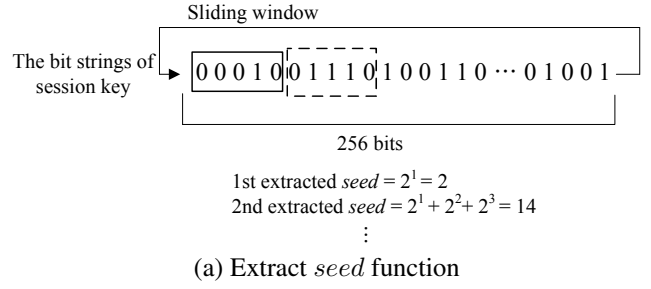


Figure 2. Encrypted message



If extracted  $seed = 2$

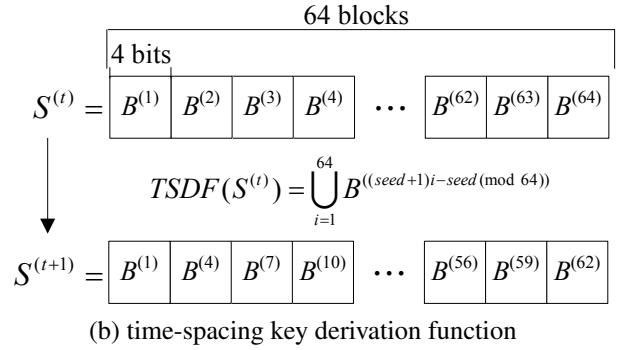


Figure 3. *ESF* and *TSDF*

Table 2. Cost of key establishment

	The number of		
	MAC	XOR	communication
Sender	2/1	3/1	1/1
Receiver	2/1	3/1	1/0
* initial key establishment / key update phase			

is more efficient than other ciphers using several permutations and substitutions in not only computational cost but also memory space consumption.

## 6. Conclusions

This paper proposed light weight secure protocols for data transmission in WSNs. These proposed protocols support key establishment and data encryption of communication nodes. For key establishment, only *MAC* and *XOR* operations are used in an efficient manner and in the initial key establishment phase, communications costs are saved because of broadcasting method. For data encryption, a new encryption method providing efficiency and security from using *XOR* operation and time spacing key derivation function was proposed. These proposed protocols are also satisfied at security aspects, which are confidentiality, authentication, integrity and freshness.

It is expected that proposed protocols have a close relationship with various environments and applications. From

this point of view, practical testing and simulations are needed and more studies about the specific nature of wireless sensor networks are needed.

## Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments in improving our manuscript. This research was supported by 2<sup>nd</sup> Brain Korea 21 Project in 2008 and the Korea Science and Engineering Foundation(KOSEF) grant funded by the Korea government(MOST). (No. R01-2006-000-10614-0)

## References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [2] H. Chan, A. Perrig, and D. X. Song, "Random key predistribution schemes for sensor networks," pp. 197–214, 2003.
- [3] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "Spins: security protocols for sensor networks," in *MobiCom '01*. ACM Press, 2001, pp. 189–199.
- [4] K. Ren, K. Zeng, W. Lou, and P. J. Moran, "On broadcast authentication in wireless sensor networks," *Lecture notes in computer science*, vol. 4138, pp. 502–514, 2006.
- [5] I. Doh and K. Chae, "Key establishment and authentication mechanism for secure sensor networks," *Advanced Web and Network Technologies, and Applications*, pp. 335–344, 2006.
- [6] E. Magkos, P. Kotzanikolaou, and M. Stefanidakis, "An asymmetric key establishment protocol for multiphase self-organized sensor networks," *Enabling Technologies for Wireless Multimedia Communications*, 2006.
- [7] P. Kotzanikolaou, E. Magkos, C. Douligeris, and V. Chrissikopoulos, "Hybrid key establishment for multiphase self-organized sensor networks," *wowmom*, vol. 03, pp. 581–587, 2005.
- [8] Y. Wu, D. Ma, T. Li, and R. H. Deng, "Classify encrypted data in wireless sensor networks," *IEEE Vehicular Technology Conference*, vol. 60, no. 5, pp. 3236–3239, 2005.
- [9] S. M. D. N. P. Cam, H.; Ozdemir, "Energy efficient security protocol for wireless sensor networks," *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, vol. 5, pp. 2981–2984, 6-9 Oct. 2003.
- [10] Y. Wang, "Robust key establishment in sensor networks," *SIGMOD Rec*, vol. 33, no. 1, pp. 14–19, 2004.
- [11] B.-C. C. Lai, D. D. Hwang, S. P. Kim, and I. Verbauwhede, "Reducing radio energy consumption of key management protocols for wireless sensor networks," in *ISLPED '04*. ACM Press, 2004, pp. 351–356.
- [12] R. Rivest, "The rc5 encryption algorithm," *Proceedings of the 2nd Workshop on Fast Software Encryption (LNCS 1008)*, pp. 86–96, 1995.
- [13] *TinyOS 2.0 Documentation*, 2006. [Online]. Available: <http://www.tinyos.net/tinyos-2.x/doc/>
- [14] R. Moskowitz, *Weakness in Passphrase Choice in WPA Interface*, 11 2003. [Online]. Available: <http://wifinetnews.com/archives/002452.html>