Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 608380, 8 pages
http://dx.doi.org/10.1155/2013/608380

Research Article

Secure and Lightweight Key Distribution with ZigBee Pro for Ubiquitous Sensor Networks

Kyung Choi,[1] Mihui Kim,[2] and Kijoon Chae[1]

[1] *Department of Computer Science and Engineering, Ewha Womans University, 52 Ewhayeodae-gil, Seodaemun-gu, Seoul 120-750, Republic of Korea*
[2] *Department of Computer Engineering, Hankyong National University, 327 Chungang-no, Anseong-si, Gyeonggi-do 456-749, Republic of Korea*

Correspondence should be addressed to Kijoon Chae; kjchae@ewha.ac.kr

Received 6 December 2012; Revised 19 June 2013; Accepted 19 June 2013

Academic Editor: Carlos Ramos

We propose a secure and lightweight key distribution mechanism using ZigBee Pro for ubiquitous sensor networks. ZigBee consumes low power and provides security in wireless sensor networks. ZigBee Pro provides more security than ZigBee and offers two security modes, standard security mode and high security mode. Despite high security mode, ZigBee Pro has weakness of key distribution. We use enhanced ECDH for secure key distribution in high security mode. Our simulation results show that the energy consumption of our approach decreases and the average run time is decreased by 39%. Moreover, the proposed scheme enhances security, that is, confidentiality, message authentication, and integrity. We also prove that the proposed key distribution can resist man-in-the-middle attack and replay attack.

1. Introduction

Various sensors in a sensor network technology are located within wired/wireless network infrastructures. Spatially distributed autonomous sensors monitor physical or environmental conditions such as temperature, humidity, sound, vibration, pressure, and motion and pass their data through the wired/wireless network to a base station. Sensor network technology has been utilized in monitoring military, home automation, and health care systems, as well as agriculture and weather conditions.

Sensors have limited memory and throughput capacity for wireless sensor networks. Therefore, limitations of the sensor itself and the underlying vulnerability of wireless communication with the sensors must be considered. In addition, sensed and transmitted data in each field are usually private information or important authentication information. Thus, security is to be applied in most cases. For this, ZigBee [1] provides a low power consumption and security standard-based protocol for applications on wireless sensor networks. ZigBee was developed to address the following needs: low cost, security, reliablility and self-healing, flexibility and extendibility, low power consumption, being easy and inexpensive to deploy, being global with use of unlicensed radio bands, integrating intelligence for network setup and message routing.

ZigBee Pro [2] (the latest specification for ZigBee, is termed ZigBee-2007) revolves around mesh networking, enhancing security. ZigBee Pro [3] also supports a large number of interoperable standards, including ZigBee health care, ZigBee home automation, ZigBee remote control, ZigBee smart energy, ZigBee telecom services, ZigBee building automation, ZigBee input device, ZigBee light link, ZigBee network devices, and ZigBee retail services. The ZigBee home automation profile [4] for a smart home allows consumers to save money, be more environmentally aware, feel more secure, and enjoy a variety of conveniences that make homes easier and less expensive to maintain.

However, the enhanced key management mechanism still has vulnerabilities in key distribution. The ZigBee home automation profile has just been applied at the network level. An enhanced mechanism is needed for this.

CertiCom [5] issues ZigBee Smart Energy certificates to manufactures whose products are certified by the ZigBee Alliance. The ZigBee Smart Energy PKI uses Elliptic Curve Qu Vanstone (ECQV) implicit certificates, which serve as an identity certificate for each ZigBee Smart Energy device. However, it does not improve ZigBee Pro itself but uses PKI separately.

In this paper, we apply the ECDH (Elliptic Curve Diffie-Hellman) [6] key distribution mechanism for ZigBee Pro vulnerabilities and propose a more efficient ECDH using sub-MAC [7] that has message authentication and prevents man-in-the-middle attack and replay attack. Our research exhibits an enhanced key mechanism and message authentication in ZigBee Pro for ubiquitous sensor networks.

The rest of this paper is organized as follows. Section 2 presents related works. Section 3 presents the proposed enhanced key distribution mechanisms. Section 4 illustrates simulation environments and analyzes the simulation to evaluate the effectiveness of our scheme. In Section 5, we analyze our approach from the view point of security. Finally, we conclude this paper in Section 6.

Table 1: Security keys.

	Layer	Msg. type	Creation
Master Keys	Application layer	Unicast	Key transport, Preinstallation
Network Keys	Application/ Network layer	Broadcast	Key transport, Preinstallation
Link Keys	Application layer	Unicast	Key transport, Preinstallation Key establishment (using Master Key)

Table 2: Security modes.

Feature	Standard	High
Network layer security provided using a Network Key	∨	∨
APS layer security provided using Link Keys	∨	∨
Centralized control and update of keys	∨	∨
Ability to switch from active to secondary keys	∨	∨
Ability to derive Link Keys between devices		∨
Entity authentication and permissions table supported		∨

2. Related Work

2.1. ZigBee Pro. ZigBee Pro is a standard specified in ZigBee-2007. ZigBee Pro improves the security of the ZigBee 2006 version with two new security modes: standard security mode compatible with the residential security of ZigBee-2006- and high security mode compatible with the commercial security of ZigBee 2006.

ZigBee security, which is based on a 128 bit AES (Advanced Encryption Standard) [8] algorithm, adds to the security model provided by IEEE 802.15.4. The security services of ZigBee include methods for key establishment and transport, device management, and frame protections. ZigBee uses three types of keys to manage security: Master, Network, and Link.

Master Keys are used as an initial shared secret between two devices, when they perform the key establishment procedure (SKKE) to generate Link Keys. Keys that originate from the Trust Center are termed Trust Center Master Keys, while all other keys are termed Application Layer Master Keys. Network Keys perform security for the Network Layer on a ZigBee network. All devices on a ZigBee network share the same key. High Security Network Keys must always be sent encrypted over the air, while Standard Security Network Keys can be sent either encrypted or unencrypted. Link Keys as an optional key secure unicast messages between two devices at the Application Layer. Keys that originate from the Trust Center are termed Trust Center Link Keys, while all other keys are termed Application Layer Link Keys. Table 1 summarizes the security keys.

ZigBee Pro offers two different security modes (i.e., Standard and High) and features as shown in Table 2 [9].

In the standard security mode, the list of devices, Master Keys, Link Keys, and Network Keys, can be maintained either by the Trust Center or by the devices themselves. The Trust Center is still responsible for maintaining a Standard Network Key; it controls policies of network admittance. In the high security mode, the Trust Center maintains a list of devices, Master Keys, Link Keys, and Network Keys that it needs to control and enforce the policies of Network Key updates and network admittance.

Unlike standard security mode, high security mode supports the ability to derive Link Keys between devices and entity authentication and permissions table supported. The Master Keys and Network Keys are preinstalled or transported; Link Keys are used in key-establishment based on a Master Key. Unencrypted key transport will give rise to serious security vulnerability [10].

2.2. ECDH. ECDH [6] is a key exchange algorithm, the well-known Diffie Hellman [11] key agreement based on ECC (Elliptic Curve Cryptography) [12]. ECDH is important in modern protocols as a key exchange and can be adopted for ECC. Figure 1 shows the key exchange process.

Consider two parties, A and B, willing to exchange a common secret key. Both have agreed to a common and publicly known curve E over a finite field, as well as to a base point Q. User A randomly chooses k_A, $1 < k < 2^Q$ and User B accordingly k_B, $1 < k < 2^Q$. User A computes a public key $Q_A = k_A Q$, User B does $Q_B = k_B Q$. User A sends Q_A to User B, User B sends Q_B to User A. User A computes the shared secret key by $P = k_A Q_B$ and User B also by $P = k_B Q_A$ [13]. An eavesdropper knows only Q_A and Q_B but is unable to compute the secret key from this. However, vulnerability of ECDH has no authentication [14] and no prevention of man-in-the-middle attack [15].

3. Proposed ZigBee Key Distribution

3.1. Standard Security Mode. The transport-key command sent from the Router to the Joiner shall not be secured in

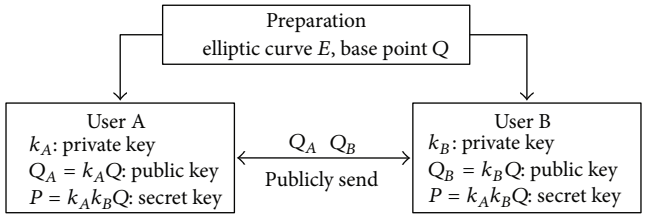

Figure 1: ECDH.

standard security mode. For this, we apply ECDH for secure Network key generation and transmission and sub-MAC mechanism for message authentication and integrity. We proved that our scheme could provide efficiency by achieving a similar run time and similar energy consumed in standard security mode [16].

3.2. High Security Mode. If the Trust Center does not already share a Master or Link Key with the newly joined device, Figure 2 shows the high security mode authentication procedure of ZigBee Pro.

The Symmetric-Key Key Establishment (SKKE) protocol is a process in which an initiator device (Trust Center) establishes a Link Key with a responder device (Joiner) using a Master Key. The next step is an entity authentication process between Router and Joiner.

As in standard security mode, Update-Device Command and Secured Transport-Key Command are encrypted with Master key, but Transport-Key Command sent from the Router to the Joiner is not secure. This has a security issue.

The MAC scheme is used for key confirmation in SKKE. The first 128 bits of keying data shall be a Mac Key and the second 128 bits shall be a Link Key during Mac Key generation. After SKKE, the Network Key is securely transmitted using the Master Key.

We propose a procedure to ensure key secure distribution as shown in Figure 3.

Trust Center $\rightarrow$ Joiner: J, aQ, N_S, sub-MAC(aQ, N_S, J)

(i) J: Joiner's 64-bit address

(ii) aQ: Trust Center generates value for key

(iii) N_S: nonce value

(iv) Sub-MAC(aQ, N_S, J): sending message sub-MAC.

When the Trust Center receives an APSME-UPDATE-DEVICE.request message, the Trust Center generates an aQ for secure Master Key and nonce N_S, and sends J, aQ, N_S, sub-MAC(aQ, N_S, J) to the Joiner. The Joiner generates sub-MAC(aQ, N_S, J) to compare the transmitted sub-MAC(aQ, N_S, J). If they match, the Joiner confirms that the transmitted message has not been modified. Otherwise, the Joiner discards the transmitted message. If the check is successful, the Joiner computes $K' = abQ$, and computes K using the Matyas-Meyer-Oseas (MMO) hash function [17]. The 160-bit K' becomes a 128 bit Network Key, K.

A sub-MAC [7] is constructed by selecting some bits of an HMAC. We reduce the overhead by transmitting only a part of the actual HMAC, rather than the entire HMAC using sub-MAC. Sub-MAC guarantees message integrity and authentication. Our research selects 8-bits of 16 bytes. We assume each node has the same PRNG (Pseudo Random Number Generator) [18].

Joiner $\rightarrow$ Trust Center: TC, bQ, N_{S+1}, sub-MAC(Master K)

(i) TC: Trust Center's 64-bit address

(ii) bQ: Joiner generates value for key

(iii) N_{S+1}: add 1 to transmitted nonce

(iv) Sub-MAC(Master K): sub-MAC using Master Key.

The Joiner sends bQ, N_{S+1}, and sub-MAC(Master K) to the Trust Center the Trust Center computes K', Master Key K = MMO(K'), and then computes sub-MAC(K) to check message integrity and computation accuracy.

Trust Center $\rightarrow$ Joiner: $E_K(N_{S+1})$

(i) $E_K(N_{S+1})$: encrypt N_{S+1} with Master Key.

Next, the generated Master Key encrypts N_{S+1}, and the result, $E_K(N_{S+1})$ is sent to the Joiner to check message integrity and announce successful Master Key generation. The Joiner decrypts the $E_K(N_{S+1})$ with the Master Key and checks the N_{S+1} to verify secure Master Key generation. If successful, the Trust Center and the Joiner perform the next step, SKKE, to establish a Link Key.

4. Simulation and Results

The Qualnet simulator was used to evaluate the performance of the proposed scheme. Our research uses Qualnet 4.5 [19] with sensor network libraries based on the ZigBee protocol and additional protocols.

We composed one clustering network structures. The clusters were composed of 15 nodes. Node 1 is a Joiner, node 16 is a Router, and node 8 is a Trust Center.

4.1. Efficiency Analysis of Enhanced Key Mechanism. We propose an enhanced key distribution scheme using ECDH for secure and lightweight key distribution and sub-MAC to overcome the vulnerability of ECDH. The simulation was performed ten times in each of the previous four procedures with Trust Center, Router, and Joiner.

First, we performed the key generation in standard security mode and high security mode, proposed key distribution in standard mode (Standard_ECDH), and proposed key distribution in high security mode (High_ECDH). Figure 4 shows the total run time measurements.

The average run time of the standard security mode is 0.5156 seconds, and for proposed key distribution in standard mode (Standard_ECDH) it is 0.5778 seconds; the difference is 0.0622 seconds. When this value is compared to the average run time of standard security mode, it adds 12%. However, the difference, 0.0622, is slight in terms of the figure and compared to the enhanced security.

The average run time of high security mode is 1.078; the average run time of proposed key distribution in high security mode (High_ECDH) is 0.6563; it decreases 0.4217. When this

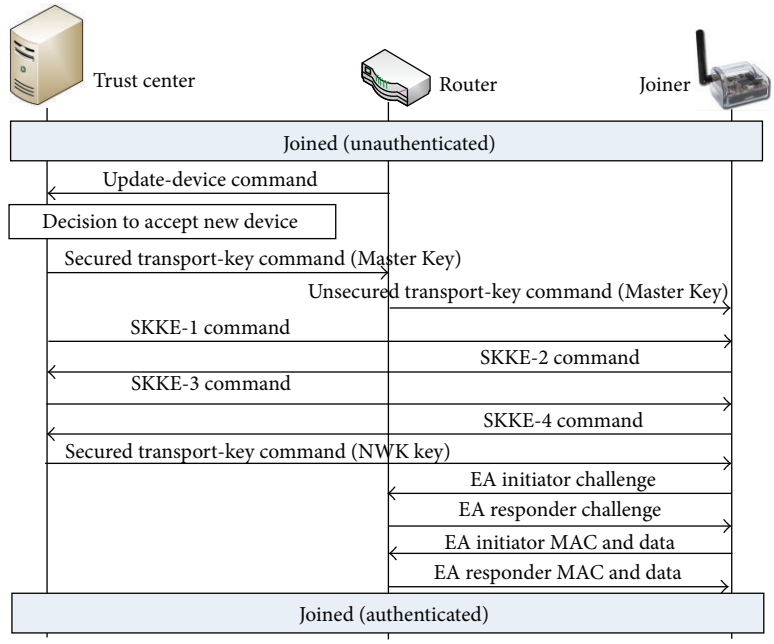

Figure 2: High security mode authentication procedure.

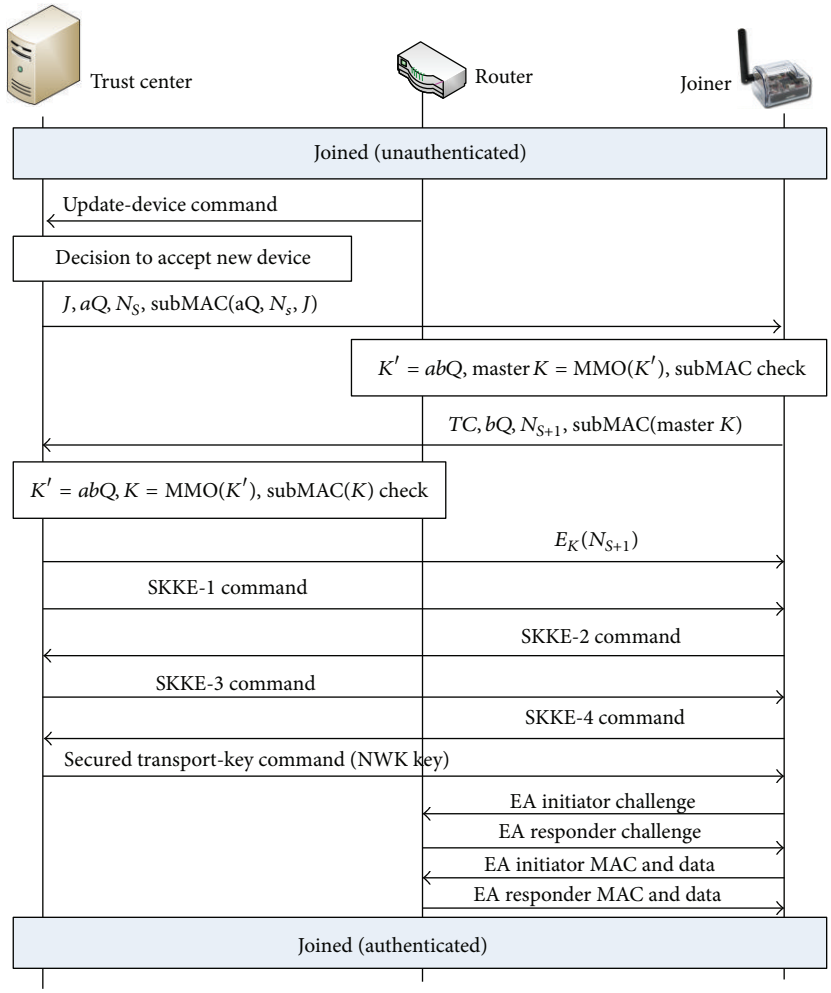

Figure 3: Proposed key distribution in high security mode.

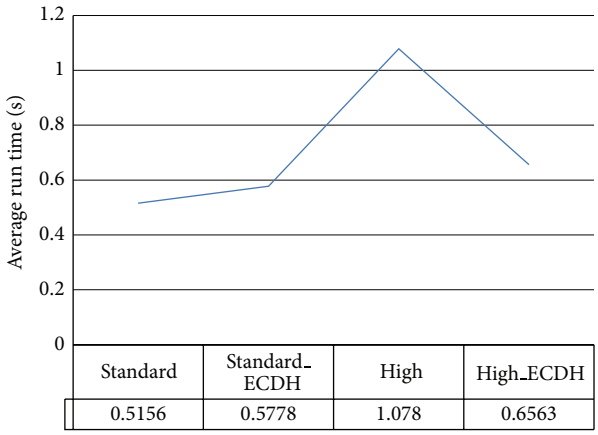

Figure 4: Simulation result-run time.

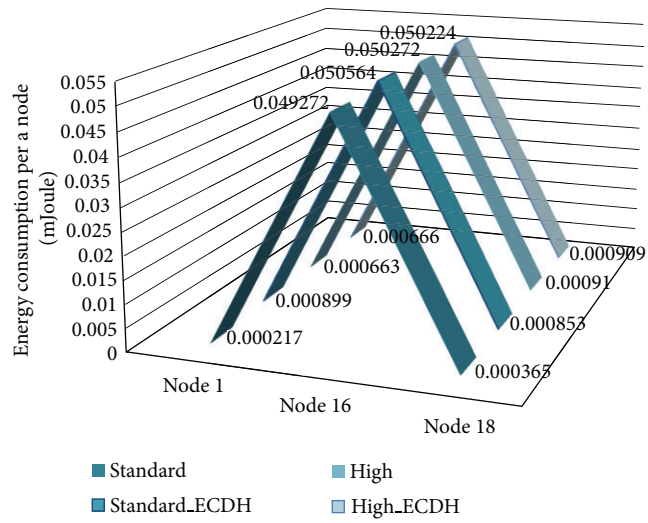

Figure 5: Energy consumed in transmit mode.

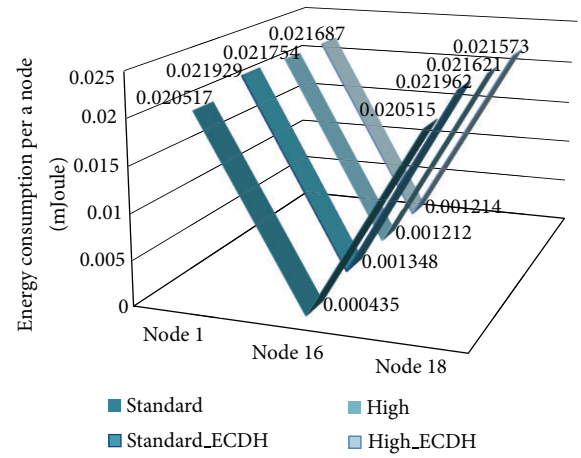

Figure 6: Energy consumed in receive mode.

value is compared to the average run time of high security mode, it is decreased by 39%. It also provides enhanced security.

Next, we measured energy consumption in Joiner (Node 1), Router (Node 16), and Trust Center (Node 18). Figure 5 shows average energy consumption in transmit mode. Figure 6 shows average energy consumption in receive mode. The average energy consumption of each node for transmit mode and receive mode is similar.

Table 3 details the values. When the proposed key distribution in security mode is compared to the standard security mode, it consumes more energy. Especially, the receive mode of the Trust Center (N18-R) shows the maximum difference, 0.001447 mJoule. However, the Trust Center has sufficient capacity and energy, so this difference is negligible. The second difference is 0.001412 mJoule in the receive mode of the Joiner (N1-R). The sensor node uses two AA alkaloid batteries. An AA alkaloid battery contains a maximum of 3000 mAh, so the total energy is 6000 mAh. The formal voltage of an AA battery assumes 1.5 volts. The amount of eletric power is 9 Wh, products of 6 Ah and 1.5 V, and this is converted into 32,400 J, 3600 X 9 (J) [20]. The difference is slight compared to 32,400 J.

The energy consumption of the high security mode and proposed key distribution in high security mode (High_ECDH) is similar. The energy consumption of proposed key distribution in high security mode (High_ECDH) decreases, except for the transmit mode of the Joiner (N1-T) and the receive mode of the Router (N16-R). Moreover, the proposed scheme enhances security.

Table 3: Energy consumption.

	STANDARD	Stand._ECDH	High	High_ECDH
N1-T	0.000217	0.000899	0.000663	0.000666
N1-R	0.020517	0.021929	0.021754	0.021687
N16-T	0.049272	0.050564	0.050272	0.050224
N16-R	0.000435	0.001348	0.001212	0.001214
N18-T	0.000365	0.000853	0.00091	0.000909
N18-R	0.020515	0.021962	0.021621	0.021573

5. Security Analysis

In this section, we analyze our enhanced key distribution for ZigBee Pro that provides security properties and resists some general attacks. ZigBee Pro is vulnerable in the case of key distribution in two security modes. ECDH cannot prevent man-in-the-middle attack and does not provide authentication. However, our proposed scheme overcomes these vulnerabilities and enhances security. Our scheme could resist man-in-the-middle attack, replay attack, and ensure confidentiality of keys, message authentication, and message integrity [16].

We assume that an attacker does not know the sub-MAC method. Therefore, even if the attacker knows the Joiner's private key b, he/she cannot make the sub-MAC message. If the attacker tries to make the sub-MAC message, the probability of failure enhances because the attacker does not know how to create a sub-MAC message using Master Key. Additionally, there is a public key infrastructure (PKI)

system. The Trust Center assures the private key b using the received public key bQ through a certificate authority (CA).

The security of a MAC scheme can be quantified in terms of the success probability achievable as a function of total number of queries to forge the MAC [21]. The security of a i-byte MAC is quantified as $2^{(i\times 8)}$ because an intruder has a 1 in $2^{(i\times 8)}$ chance in blindly forging the MAC. To increase the security of a MAC, its size should be increased. Increasing the size of the MAC also increases the communication overhead [22]. Our sub-MAC selects 8 bits of 128 bits. Therefore, the security of the sub-MAC is 28. Hence, the possibility that the false data are not detected by a sub-MAC is $1/2^8$ (=0.0039). Moreover, the communication overhead is reduced by 1/16 (=0.0625). Consequently, the size of the sub-MAC is directly related to the strength of the security and the communication overhead. A balance needs to be achieved between the desired security level and the transmission overhead [7].

5.1. BAN Analysis. BAN logic (the Logic of Authentication of Burrows, Abadi and Needham) [23] is widely used and studied in formal analysis due to its simplicity and efficiency. The BAN logic is a model logic based on belief and can be used in the analysis and design of a cryptographic protocol. The use of a formal language in the analysis and design process can exclude faults and improve the security of the protocol.

5.1.1. Basic Notations. The symbols A, B, P, and Q are principals involved in this sort of key agreement protocol: K_{AB} represents a good session key for communication between A and B [24].

$P| \equiv X$: Principal P believes X. P believes as if X is true.

$P \triangleleft X$: P sees X. A principal has sent P a message containing X.

$P| \sim X$: Principal P once said X. P at some time believed X and sent it as part of a message.

$P \Rightarrow X$: Principal P has jurisdiction over X. Principal P has authority over X and is trusted in this matter.

$\#(X)$: The formula X is fresh. That is, X has not been sent in a message at any time before the current run of the protocol. A message that is created for the purpose of being fresh is called a nonce.

$P \overset{K}{\longleftrightarrow} Q$: P and Q may use a shared key K to communicate. The key is good and will always be known only to P and Q and to any other principal trusted by either of them.

$\{X\}_K$: X is encrypted using key K.

5.1.2. Inference Rules. Message Meaning Rules for shared keys:

$$\frac{P \mid\equiv P \overset{K}{\longleftrightarrow} Q,\ P \triangleleft \{X\}_K}{P \mid\equiv Q \mid\sim X}. \tag{1}$$

If principal P believes that key K is shared only with principal Q and sees a message X encrypted under a key K, it believes only with principal Q. P may conclude that it was originally created by Q who once said its contents.

Jurisdiction Rule is as Follows

$$\frac{P \mid\equiv P \Longrightarrow Q,\ P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}. \tag{2}$$

If P believes that Q believes X and also believes that Q has jurisdiction over X, then P should believe X too.

Nonce Verification Rule is as Follows:

$$\frac{P \mid\equiv \#(X),\ P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}. \tag{3}$$

If P believes that X is fresh and that Q once said X, then P believes that Q has said X during the current run of protocol and hence that Q believes X at present. In order to apply this rule, X should not contain any encrypted text. The nonce verification rule is the only way of "promoting" once said assertion to actual belief.

5.2. BAN Analysis of the Proposed Key Distribution

Initialization Hypothesis is as Follows

(1) Trust Center $\mid\equiv$ TC.

(2) Trust Center $\mid\equiv$ Joiner $\mid\equiv J$.

(3) Trust Center $\mid\equiv$ Joiner $\Rightarrow aQ$.

(4) Trust Center $\mid\equiv$ sub-MAC.

(5) Joiner $\mid\equiv J$.

(6) Joiner $\mid\equiv$ Trust Center $\mid\equiv$ TC.

(7) Joiner $\mid\equiv$ Trust Center $\Rightarrow bQ$.

(8) Joiner $\mid\equiv$ sub-MAC.

Proposed Key Distribution Idealization

(1) Trust Center $\rightarrow$ Joiner: J, aQ, N_S, and sub-MAC(aQ, N_S, J).

(2) Joiner $\rightarrow$ Trust Center: TC, bQ, N_{S+1}, and sub-MAC(K).

(3) Trust Center $\rightarrow$ Joiner: $E_K(N_{S+1})$.

Goal

Trust Center $\mid\equiv$ Trust Center $\overset{K}{\longleftrightarrow}$ Joiner.

Joiner $\mid\equiv$ Trust Center $\overset{K}{\longleftrightarrow}$ Joiner.

Trust Center $\mid\equiv$ Joiner $\mid\equiv$ Trust Center $\overset{K}{\longleftrightarrow}$ Joiner.

Joiner $\mid\equiv$ Trust Center $\mid\equiv$ Trust Center $\overset{K}{\longleftrightarrow}$ Joiner.

Analysis. Through the proposed key distribution idealization (1), we can get

$$\frac{\text{Joiner} \triangleleft J, aQ, N_S, \text{sub-MAC}(aQ, N_S, J), \text{Joiner} \mid\equiv \text{sub-MAC}}{\text{Joiner} \mid\equiv J, aQ, N_S},$$

$$\frac{\text{Joiner} \mid\equiv aQ}{\text{Joiner} \mid\equiv K}. \tag{4}$$

Through the proposed key distribution idealization (2), we can get

$$\text{Trust Center} \triangleleft \text{TC}, bQ, N_{S+1}. \tag{5}$$

The Trust Center computes K and then sub-MAC(K) as follows:

$$\frac{\text{Trust Center} \triangleleft \text{sub-MAC}(K), \text{Trust Center} \mid\equiv \text{sub-MAC}}{\text{Trust Center} \mid\equiv K},$$

$$\frac{\text{Trust Center} \mid\equiv K}{\text{Trust Center} \mid\equiv bQ, \text{Trust Center} \mid\equiv \#(N_{S+1})},$$

$$\frac{\text{Trust Center} \mid\equiv K}{\text{Trust Center} \mid\equiv \text{Trust Center} \overset{K}{\longleftrightarrow} \text{Joiner}}. \tag{6}$$

And then, Trust Center $\mid\equiv$ Joiner $\mid\equiv$ Trust Center $\overset{K}{\longleftrightarrow}$ Joiner.

Through the proposed key distribution idealization (3), we can get

$$\frac{\text{Joiner} \triangleleft \{N_{S+1}\}_K}{\text{Joiner} \mid\equiv \text{Truster Center} \overset{K}{\longleftrightarrow} \text{Joiner}}. \tag{7}$$

And then, Joiner $\mid\equiv$ Trust Center $\mid\equiv$ Trust Center $\overset{K}{\longleftrightarrow}$ Joiner.

According to the formalization analysis, we can get the conclusion that the proposed key distribution can resist man-in-the-middle-attack and replay attack.

6. Conclusion

This work proposed an enhanced key distribution scheme using ECDH and sub-MAC for efficiency and security. We have applied ECDH for secure key distribution and improved vulnerability of ECDH, using sub-MAC and nonce for message freshness and integrity.

We compared ZigBee Pro to the proposed scheme. We proved that our scheme could provide efficiency by achieving a shorter run time and lower energy consuming in high security mode. Security analysis proved our scheme could resist man-in-the-middle attack, replay attack, and provide confidentiality, message authentication, and integrity. Consequenly, the proposed scheme provides lightweight and secure key distribution compared to ZigBee Pro. We are going to experiment our proposed scheme with ZigBee devices in future work.

Acknowledgments

The work was supported by Ewha Global Top 5 Grant 2011 of Ewha Womans University and World Class University Program (R33-10085) through National Research Foundation of Korea funded by the Ministry of Education, Science and Technology. It was also in part supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0014020).

References

[1] IEEE Std 802.15.4-2003, "Wireless Medium Access Control and Physical Layer Specifications for Low-Rate Wireless Personal Area Networks," IEEE, 2003.

[2] ZigBee Alliance, "ZigBee-2007 Specification," January 2008.

[3] http://www.zigbee.org/Standards/Overview.aspx.

[4] ZigBee Alliance, "ZigBee Home Automation Public Application Profile," ZigBee Document 053520r26, February 2010, http://zigbee.org/Standards/ZigBeeHomeAutomation/download.aspx.

[5] http://www.certicom.com/index.php/device-authentication-service/smart-energy-device-certificate-service.

[6] Certicom, "Standards for Efficient Cryptography, SEC 1: Elliptic Curve Cryptography," Ver 1.0, September 2000, http://www.secg.org/download/aid-385/sec1_final.pdf.

[7] H. Çam, N. Challa, and M. Sikri, "Secure and efficient data transmission over body sensor and wireless networks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2008, Article ID 291365, 18 pages, 2008.

[8] Advanced Encryption Standard, FIPS 197, November 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[9] Daintree Networks Inc., "Getting Started with ZigBee and IEEE 802.15.4," February 2008.

[10] C. Alcaraz and J. Lopez, "A security analysis for wireless sensor mesh networks in highly critical systems," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 40, no. 4, pp. 419–428, 2010.

[11] W. Diffie and M. E. Hellman, "New direction in cryptography," *IEEE Transactions on Information Theory*, vol. IT-22, no. 6, pp. 644–654, 1976.

[12] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, 1987.

[13] E. Blaß and M. Zitterbart, "Efficient implementation of elliptic curve cryptography for wireless sensor networks," TeleMatics Technical Report, 2005.

[14] G. De Meulenaer, F. Gosset, F.-X. Standaert, and O. Pereira, "On the energy cost of communication and cryptography in wireless sensor networks," in *Proceedings of the 4th IEEE International Conference on Wireless and Mobile Computing, Networking and Communication (WiMob '08)*, pp. 580–585, October 2008.

[15] T. Chung and U. Roedig, "DHB-KEY: an efficient key distribution scheme for wireless sensor networks," in *Proceedings of the 5th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '08)*, pp. 840–846, October 2008.

[16] K. Choi, M. J. Yoon, M. H. Kim, and K. J. Chae, "An enhanced key management using ZigBee Pro for wireless sensor networks," in *Proceedings of the 26th International Conference on Information Networking (ICOIN '12)*, Bali, Indonesia, February 2012.

[17] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

[18] D. Seetharam and S. Rhee, "An efficient pseudo random number generator for low-power sensor networks," in *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN '04)*, pp. 560–562, Tampa, Fla, USA, November 2004.

[19] QualNet 4. 5, Scalable Network Technologies, Inc., http://www.scalable-networks.com/.

[20] K. Choi, M.-H. Kim, K.-J. Chae, J.-J. Park, and S.-S. Joo, "An efficient data fusion and assurance mechanism using temporal and spatial correlations for home automation networks," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, pp. 1330–1336, 2009.

[21] P. Gauravaram, W. Millan, J. G. Nieto, and E. Dawson, "3C-A provably secure pseudorandom function and message authentication code: a new mode of operation for cryptographic hash function," Cryptology ePrint Archive, Rep., 2005.

[22] S. Ozdemir and H. Çam, "Integration of false data detection with data aggregation and confidential transmission in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 736–749, 2010.

[23] M. Burrows, M. Abadi, and R. Needham, "Logic of authentication," *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18–36, 1990.

[24] S. Yang and X. Li, "A limitation of BAN logic analysis on an man-in-the-middle attack," *Journal of Information and Computing Science*, vol. 1, no. 3, pp. 131–138, 2006.