Hindawi Publishing Corporation
Journal of Sensors
Volume 2016, Article ID 2678269, 9 pages
http://dx.doi.org/10.1155/2016/2678269

Hindawi

Research Article

Power Adaptive Data Encryption for Energy-Efficient and Secure Communication in Solar-Powered Wireless Sensor Networks

Jong Min Kim, Hong Sub Lee, Junmin Yi, and Minho Park

School of Electronic Engineering, Soongsil University, 369 Sangdo-ro, Dongjak-gu, Seoul 06987, Republic of Korea

Correspondence should be addressed to Minho Park; mhp@ssu.ac.kr

Received 16 November 2015; Revised 5 February 2016; Accepted 11 February 2016

Academic Editor: Hana Vaisocherova

Basic security of data transmission in battery-powered wireless sensor networks (WSNs) is typically achieved by symmetric-key encryption, which uses little energy; but solar-powered WSNs sometimes have sufficient energy to achieve a higher level of security through public-key encryption. However, if energy input and usage are not balanced, nodes may black out. By switching between symmetric-key and public-key encryption, based on an energy threshold, the level of security can be traded off against the urgency of energy-saving. This policy can also reduce the amount of energy used by some nodes in a WSN, since data encrypted using a public-key is simply relayed by intermediate nodes, whereas data encrypted using a symmetric-key must be decrypted and reencrypted in every node on its path. Through a simulation, we compared the use of either symmetric-key or public-key encryption alone with our scheme, which was shown to be more secure, to use energy more effectively, and to reduce the occurrence of node blackouts.

1. Introduction

Wireless sensor networks (WSNs) are increasingly being used in smart homes, by the military, for disaster detection, and in other applications which require the monitoring of environments. WSNs consist of many wireless sensor nodes that collect data and a sink node that receives the data from the sensor nodes. Then, the sink forwards that data to the server for analysis through Internet. Most wireless sensor nodes are battery-powered and therefore have a limited lifetime. There has been a lot of research on reducing the energy consumption of nodes in order to increase network lifetime. One approach is to employ nodes which harvest energy from the environment. Solar-powered nodes have especially been preferred because of the high density and relatively stable supply of sunlight.

Meanwhile, many WSNs are deployed outside buildings, and the nodes in these networks are vulnerable to attack. Data may be robbed or spurious data may be introduced; DoS attacks may seek to exhaust the energy in the nodes [1]. Encryption techniques are widely used to address these problems in other networks. However, WSNs tend not to use encryption techniques because of energy concerns, or they use just symmetric-key-based encryption techniques tailored to WSNs. These consume less energy but show lower levels of encryption than other encryption schemes used in personal devices such as laptops or desktop computers.

As shown in Figure 1, there is a trade-off relationship between the lifetime and the level of data security in WSNs. Though the encryption scheme provides a higher level of security, more energy is required; thus the lifetime of the WSN decreases. One approach to solve this problem is to utilize an environmental energy source instead of a battery. Among the various environmental energy sources, this paper focuses on solar energy.

In this work, we propose an energy-aware security level control scheme (ESCS), which is the extended version of our previous work [2]. It is designed to increase the level of security in WSNs by using only surplus energy at each energy harvesting node. This means that there is no trade-off anymore between the WSN lifetime and the level of security, since the proposed scheme utilizes only surplus energy. A node which has more energy in its battery than a certain threshold encrypts the data by using a public-key-based encryption scheme. The notable impact of this operation is not only strengthening data security but also mitigating the energy consumption of relaying nodes. When the amount of energy remaining drops below a certain threshold, the node

Table 1: Summary of power consumption of commercial sensor network nodes.

	Crossbow MICAz [33]	Intel IMote2 [34]	Jennic JN5139 [35]
Radio standard	IEEE 802.15.4/ZigBee	IEEE 802.15.4	IEEE 802.15.4/ZigBee
Typical range	100 m (outdoors), 30 m (indoors)	30 m	1 km
Data rate (kbps)	250 kbps	250 kbps	250 kpbs
Sleep mode (deep sleep)	15 μA	390 μA	2.8 μA (1.6 μA)
Processor only	8 mA active mode	31~53 mA*	2.7 + 0.325 mA/MHz
RX	19.7 mA	44 mA	34 mA
TX	17.4 mA (+0 dbm)	44 mA	34 mA (+3 dBm)
Supply voltage (minimum)	2.7 V	3.2 V	2.7 V
Average	2.8 mW	12 mW	3 mW

*Consumption depends on clock speed selected between 13 and 104 MHz.

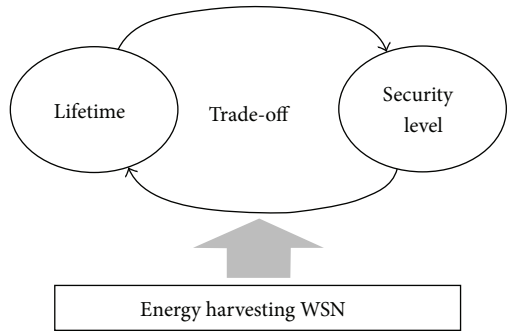

Figure 1: Trade-off relationship between the lifetime and the level of security in WSN.

reduces its level of data security by using a symmetric-key-based encryption scheme. It does this, in order to reduce the possibility of a blackout.

The rest of this paper is organized as follows: in Section 2 we describe research related to solar-powered WSNs and encryption techniques. In Section 3 we introduce our energy-adaptive encryption scheme. In Section 4 we assess the performance of our technique through an experiment, and in Section 5 we draw conclusions.

2. Related Work

2.1. Energy Supply and Demand of Solar-Powered Sensor Node. Examples of the power consumption of a selection of commercial sensor network nodes for a range of operating conditions are given in Table 1 [3]. The average values given in Table 1 are based on an operating regime of communication (RX and TX, i.e., receive and transmit) for 1% of the time, processing for 10% of the time, and sleeping for the remaining time. We can confirm that the energy demand of sensor node is several mW.

Meanwhile, we will now analyze the energy supply in a solar-powered node. PV (Photovoltaic) conversion of visible light into electrical power is well established and PV devices provide relatively high efficiency over a broad range of wavelengths. These devices are typically of low cost and provide voltage and current levels that are close to those required for microelectronic circuits. The average solar insulation at the top of the Earth's atmosphere is approximately 1370 $W \cdot m^{-2}$ [4]. The energy available for harvesting at a particular location on the Earth's surface clearly varies with time of day, latitude, and atmospheric conditions and the efficiency of conversion depends on the incidence angle to the PV device. Annually averaged surface-received energy varies from around 300 $W \cdot m^{-2}$ near the equator to around 100 $W \cdot m^{-2}$ near the poles. For temperate regions, the daily average available shortwave energy varies from around 25 $MJ \cdot m^{-2} \cdot day^{-1}$ in summer to around 3 $MJ \cdot m^{-2} \cdot day^{-1}$ in mid-winter [5]. This does however depend on prevailing atmospheric conditions with heavy cloud cover resulting in a drop in available energy of approximately an order of magnitude. Given that commercially available PV cells provide a typical efficiency of around 15%, the minimum average electrical power over a 24-hour period in a temperate location is around 2 $W \cdot m^{-2}$. An important consideration in solar energy harvesting is that the energy is delivered for only part of the day and, assuming the sensor network is required to operate at the same level at all times, the energy gathered during the day must be stored for night time operation. Considering the same temperate location as considered above, a total of 0.15 $MJ \cdot m^{-2}$ electrical energy is harvested over an 8-hour period during the day in winter and must be stored to provide for the remaining 16 hours of the day. Commercially available supercapacitors have energy densities of around 5 $kW \cdot h \cdot m^{-3}$. Thus, over a 24-hour period, an average power of approximately 200 $W \cdot m^{-3}$ could be stored. This figure would correspond to an average power of 0.2 mW/cm^3. Assuming that the node uses 20 cm^3 (2 cm × 2 cm × 5 cm) PV cell, it can provide an average of 4 mW of power, and this power supply can meet the energy demand of typical sensor nodes described in Table 1.

2.2. Solar Energy Harvesting WSNs. Since solar energy has a high power density and a largely periodic availability, it has become the most attractive energy source for WSNs. Many prototypes of solar-powered WSNs have been reported [6–13]. Most of these projects have focused on aspects of node-level design, such as power control or hardware structure, and they do not address the issue of performance optimization at application or network levels, involving metrics such as data throughput or data reliability. For example, Kansal et al. [14]

explained the important issues in the design of solar energy harvesting nodes and reported the implementation of a prototype called Heliomote. Alippi and Galperti [9] designed a low power maximum-power-point tracker (MPPT) circuit to transfer energy harvested by solar panels to a rechargeable battery; Taneja et al. [10] proposed a systematic approach to the design of a micro solar energy harvesting server system in a wireless sensor node.

More recent research has considered the performance of an entire solar energy-powered sensor network [15–17]. For example, Noh and Hur [17] studied end-to-end delay of data as a measure of the quality of service (QoS) provided by a solar energy-powered sensor network. They proposed a technique for achieving the fastest data transmission, taking the locations of nodes into account, together with the amount of energy in each node and the duty-cycle of neighboring nodes. In this current study, the QoS metric is the security of data, which is to be optimized.

Yang et al. [18] suggested data distribution and resource allocation techniques to reduce data loss in solar energy-powered WSNs. The algorithm allocates nodes to layers depending on their energy levels. Nodes with abundant energy run in ER-mode (energy-rich mode) and nodes which are short of energy run in ES-mode (energy-saving mode). In ES-mode, a node gathers and stores data without transmission. In ER-mode, both the replication and transmission protocol are invoked to reduce data loss. The point is that this scheme can reduce the data loss by replication without any loss of working time. The allocation of nodes to layers is changed dynamically to reflect changes in their energy levels.

2.3. Sensor Network Security. The lack of data and energy storage and low levels of electrical and computational performance in a node represent major obstacles to the implementation of established security techniques in WSN. The unreliable communication channel and unattended operation make effective security even more difficult to achieve. It has been observed [19] that wireless sensor nodes often have the processing capabilities of computers that are decades older, and the trend in manufacturing is to reduce the cost of nodes, not to increase computing power. Nevertheless, many researchers have begun to address the challenges of WSN security given the processing energy limitations of wireless sensor nodes. The security of many different aspects of WSNs is being examined, including routing [20], data aggregation [21, 22], and group formation [23].

3. Energy-Aware Security Level Control

In this section, we introduce a scheme that effectively utilizes energy and increases the network encryption level by adaptively selecting the encryption method according to the remaining energy of energy harvesting nodes.

3.1. Symmetric-Key versus Public-Key Cryptosystem. The proposed scheme uses public-key or symmetric-key methods to encrypt sensory data according to the amount of energy. In a symmetric-key cryptosystem, the sender uses the symmetric-key to encrypt the plain text to the chipper text, and the receiver decrypts the chipper text to the plain text using the same key as the sender, as shown in

$$\begin{aligned} C &= E_K(M), \\ M &= D_K(C) = D_K\left(E_K(M)\right), \end{aligned} \tag{1}$$

where M is a plain text, C is a chipper text, K is a symmetric-key, E is an encryption function, and D is a decryption function. Since the symmetric-key algorithm uses the same key for encryption (by the sender) and decryption (by the receiver), two nodes that exchange data should maintain their own keys. In order for the receiver to relay the received data, the node decrypts the received data and reencrypts it with a key that corresponds to a new target node. Most symmetric-key algorithms perform encryption faster and consume less energy than public-key algorithms. However, because redecryption and reencryption are required for each hop, the energy consumption of other nodes that should relay that data is not inconsiderable. Furthermore, there is a problem in which the key is easily exposed because of the characteristics of WSNs where nodes are easily snatched and attacked physically. The symmetric-key encryption algorithm used in our technique is an advanced encryption standard (AES) algorithm [24].

In a public-key cryptosystem, on the other hand, a sender encrypts the plain text to the chipper text with a public-key or a private key. Then the receiver decrypts the chipper text using the private key if it was encrypted with the public-key or decrypts it using the public-key if it was encrypted with the private key, shown in

$$\begin{aligned} C &= E_{K+}(M) \text{ or } E_{K-}(M), \\ M &= D_{K+}\left(E_{K-}(M)\right) = D_{K-}\left(E_{K+}(M)\right), \end{aligned} \tag{2}$$

where M is a plain text, C is a chipper text, $K+$ is a public-key, and $K-$ is a private key, E is an encryption function, and D is a decryption function. When all nodes maintain the same public-key and transmit data which is encrypted with it, the sink node decrypts the data with its private key. At this point, the intermediate nodes that receive the data, which should be relayed to a sink node, can transmit it without reencrypting. Most public-key algorithms are CPU intensive and thus consume more energy compared to the symmetric-key algorithm. However, they show the higher reliability of data since the data (encrypted by public-key algorithm) decryption by malicious hackers or programs is nearly impossible and there is almost no risk of outflow of the key. In addition, the energy consumption for data transmission is also slightly larger than the symmetric-key method due to the greater size of the encrypted data; however, because the intermediate nodes transmit the data without reencryption, the energy consumption from the point of view of entire networks is less than the symmetric-key method. The public-key algorithm used in our technique is an Elliptic Curve Integrated Encryption Scheme (ECIES) [25].

Since all nodes should maintain private keys to use the symmetric-key algorithm in our technique, a random key predistribution technique [26] which randomly selects

Table 2: The characteristics of symmetric-key and public-key algorithm.

Algorithm	Reencryption on relaying	Encryption speed	Energy required for encoding	Security level
Public-key algorithm (ECIES)	X	Slow	More	Strong
Symmetric-key algorithm (AES)	O	Fast	Less	Weak

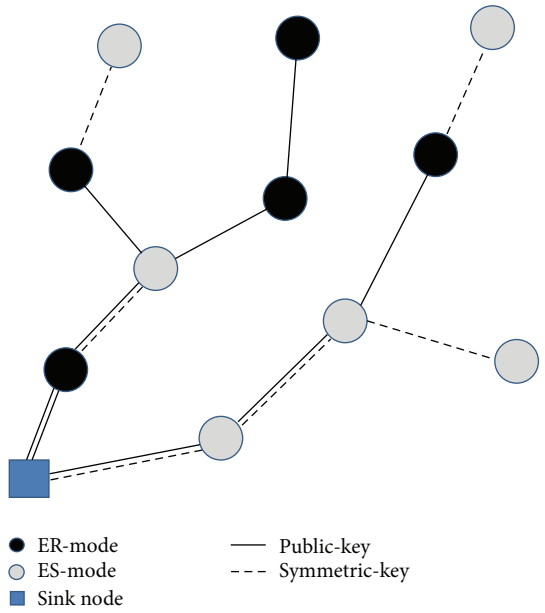

Figure 2: ESCS operation overview.

a large symmetric-key pool from the entire symmetric-key space is used to communicate with each other. Furthermore, all nodes should maintain a public-key and the sink node should maintain a private key in order to use the public-key algorithm. Table 2 compares the public-key algorithm and symmetric-key algorithm.

3.2. Overview of the Proposed ESCS (Energy-Aware Security Level Control Scheme). The proposed ESCS assumes that the WSN consists of many solar-powered nodes, which periodically collect, encrypt, and transfer data to the sink node. Each node determines its mode according to its remaining energy level and transfers data using different encryption algorithms for each mode.

Figure 2 shows a simple example. The node changes its operation in two different modes, namely, ES-mode (energy-saving mode) and ER-mode (energy-rich mode). When the remaining energy is less than a specific threshold, the node starts to operate in the ES-mode in order to focus on saving its energy instead of increasing the security level and thus encrypts data using a symmetric-key algorithm. As shown in Figure 2, the nodes in ES-mode (gray circles) decrypt and reencrypt the received packet with a symmetric-key algorithm if that packet was encrypted with a symmetric-key algorithm and just relay the packet without any decrypting or reencrypting if that packet was encrypted with a public-key algorithm. Of course, they encrypt their own sensory data with symmetric-key algorithm to save the energy.

On the other hand, when the energy remaining is more than the threshold, the node changes its mode to ER-mode and starts to focus on increasing the security level at the sacrifice of its energy because it has a sufficient amount of energy. As shown in Figure 2, the node in ER-mode (black circle) decrypts the received packet with a symmetric-key algorithm and then reencrypts it with public-key algorithm if that packet was encrypted with a symmetric-key algorithm and just relays the packet without any decrypting or reencrypting if that packet was encrypted with a public-key algorithm and encrypts its own sensory data with public-key algorithm.

We can infer from this explanation that once the data is encrypted by the public-key algorithm at the sacrifice of energy at one energy-rich node, the burden of all relaying nodes can be mitigated since they do not need to decrypt and reencrypt the data; this can lead to the decrease of energy consumption of entire network as the number of hops is increasing.

3.3. Energy Threshold Determination. First of all, let us analyze the energy consumption of a wireless sensor node. Each sensor node detects events, performs data processing, and transmits data. Therefore, energy consumption can be divided into three parts, sensing, data processing, and communication. Although the consumption from sensing depends on applications, types of sensors, detection complexity, and so forth, it is considered that the power dissipated by an Analog Digital Converter is dominant, which depends on two factors [27]:

$$p \propto F_s \cdot 2^{\mathrm{ENOB}}, \tag{3}$$

where F_s is the sampling rate and ENOB is the effective number of bits. The energy consumption in data processing depends on the clock frequency, the average capacitance, the supply voltage, thermal voltage, and so forth. A sensor node expends most energy for communication. The power consumption of communication E_C can simply be modeled as

$$E_C = E_O + E_{\mathrm{tx}} + E_{\mathrm{rx}}, \tag{4}$$

where E_O is the output transmit power and E_{tx} and E_{rx} are the power consumed in the transmitter and receiver electronics, respectively. Although all the three parts of energy consumption should be considered for the exact analysis, this work focuses only on the communication energy consumption and data processing, specifically data encryption and decryption.

We will now explore the threshold value of energy. The threshold used in mode determination is related to the energy consumption of the entire network. When the threshold is high, many nodes operate in ES-mode, security

level decreases, and the charged energy can be wasted by exceeding battery capacity. Conversely, when the threshold is low, many nodes operate in ER-mode and security level increases; however, the blackout time of some nodes increases because of an increase in node energy consumption.

Formulating an ideal energy model for a solar-powered system requires knowledge of both the energy harvesting rate of a solar-cell as an energy input model and the energy-consuming rate of the system as an energy output model. The former is dependent on the location, weather, and season where the system is deployed, and the latter is dependent on the data-sensing rate, data-transmitting rate, and duty-cycle. The problem we have to overcome is that most of these factors cannot be predicted precisely. We will now introduce a simple but effective energy model [17, 18] that is independent of these elusive factors.

Let power $P_{\text{solar}}(i)$ be the average charging rate of a solar-powered node n_i, and let $P_{\text{sys}}(i)$ be the average power consumption rate of the same node. $P_{\text{sys}}(i)$ and $P_{\text{solar}}(i)$ can be estimated when the network is operational using moving averages. Knowing the amount of energy currently available at node n_i, which we will call $E_{\text{residual}}(i)$, the expected time until the battery becomes full can be expressed as follows:

$$T_{\text{full}}\left(E_{\text{residual}}(i)\right) = \frac{C(i) - E_{\text{residual}}(i)}{P_{\text{solar}}(i) - P_{\text{sys}}(i)}, \tag{5}$$

where $C(i)$ is the battery capacity of node n_i. Note that the battery will only charge if $P_{\text{solar}}(i) > P_{\text{sys}}(i)$, which means that the average energy consumption rate of the node must be less than its average solar energy charging rate; otherwise the node would have to hibernate. Fortunately, even though $P_{\text{solar}}(i)$ cannot be controlled, $P_{\text{sys}}(i)$ can be roughly controlled by adjusting the duty-cycle DC(i) of node n_i, since $P_{\text{sys}}(i)$ is a nondecreasing function of DC(i). Therefore, by determining an upper bound on DC(i), we can fulfill inequality $P_{\text{solar}}(i) > P_{\text{sys}}(i)$.

Even though solar energy is not available at night and varies from one day to another, no blackout time would be expected between a given time and the next time the battery is full, if the amount of energy currently in the battery satisfies the following condition:

$$E_{\text{residual}}(i) \geq P_{\text{sys}}(i) \cdot T_{\text{full}}\left(E_{\text{residual}}(i)\right). \tag{6}$$

This is true even in the worst case, in which all the solar charging occurs at the very last moment, which is $T_{\text{full}}(E_{\text{residual}}(i))$. By solving (5) and (6), we can fulfill $E_{\text{residual}}(i) \geq (P_{\text{sys}}(i)/P_{\text{solar}}(i)) \cdot C(i)$. This means that the system runs without any unexpected blackouts for any pattern of weather or energy consumption, if it has at least $(P_{\text{sys}}(i)/P_{\text{solar}(i)}) \cdot C(i)$ energy in the battery. This value will be termed the energy threshold $E_{\text{threshold}}(i)$, which is formulated as a follows:

$$E_{\text{threshold}}(i) = \frac{P_{\text{sys}}(i)}{P_{\text{solar}}(i)} \cdot C(i). \tag{7}$$

To sum up, if $E_{\text{residual}}(i)$ becomes smaller than $E_{\text{threshold}}(i)$, the system cannot be guaranteed to run without an unexpected blackout time. Therefore, the node should operate in ES-mode to save energy. The other case means that the node will have enough energy to perform extra tasks such as enhancing security level by allowing it to start operating in ER-mode. Figure 3 shows the relation between the system parameters and the system status in our energy model.

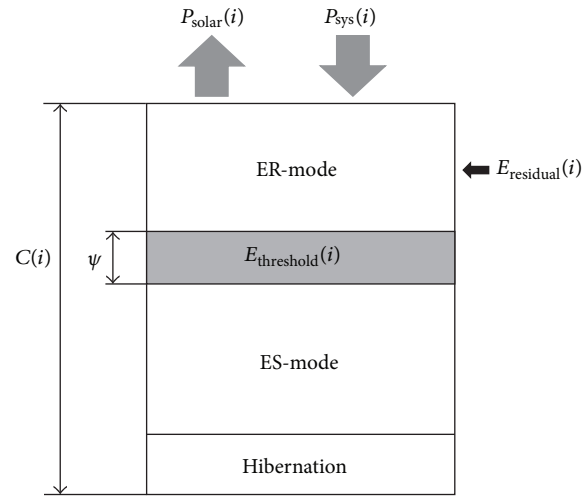

Figure 3: System parameters and system modes in our energy model.

```
(1) if E_residual(i) > E_threshold(i) then
(2)     m_i ← 1;
(3)     invoke ECIES algorithm;
(4) else
(5)     m_i ← 0;
(6)     invoke AES algorithm;
(7) end if
```

Algorithm 1: ESCS-Init(i) sudo-code.

```
(1) while 1 do
(2)     if E_residual(i) > E_threshold(i) + ψ and m_i = 0 then
(3)         m_i ← 1;
(4)         invoke ECIES algorithm;
(5)     else
(6)         if E_residual(i) < E_threshold(i) − ψ and m_i = 1 then
(7)         m_i ← 0;
(8)         invoke AES algorithm;
(9)         end if
(10)    end if
(11) end while
(12) sleep(period);
```

Algorithm 2: ESCS(i) sudo-code.

3.4. Practical Algorithm of the Proposed ESCS. Algorithms 1 and 2 summarize our scheme in a sudo algorithm with the notations presented in Table 3. In our scheme, once the nodes have been deployed, node n_i has to complete the ESCS-Init(i) to determine its initial operating mode. After

Table 3: Notation for ESCS-Init(i) and ESCS(i).

Symbol	Meaning
$E_{\text{residual}}(i)$	Amount of residual energy of node n_i
$E_{\text{threshold}}(i)$	Energy threshold for changing mode, $(P_{\text{sys}}(i)/P_{\text{solar}}(i))C$
M_i	1 if sensor node n_i is in energy-rich mode, otherwise 0
ψ	Energy window to prevent frequent change of the mode
AES	One of the symmetric-key encryption algorithms
ECIES	One of the public-key encryption algorithms
Period	Periodic invocation cycle of the ESCS(i)

ESCS-Init(i) has been run, ESCS(i) is invoked periodically at node n_i as shown in Algorithm 2. Note that it is necessary to prevent the mode of each node from changing frequently and repeatedly. Let m_i be the mode of node n_i, which is 1 if node n_i operates in ER-mode and otherwise 0. This mode m_i depends on whether $E_{\text{residual}}(i)$ is larger than $E_{\text{threshold}}(i)$. However, comparing $E_{\text{residual}}(i)$ with the exact value of $E_{\text{threshold}}(i)$ may lead to frequent changes of m_i. Suppose a node n_i starts to operate in ER-mode as soon as $E_{\text{residual}}(i)$ becomes larger than $E_{\text{threshold}}(i)$. Since node n_i has barely sufficient energy, $E_{\text{residual}}(i)$ is likely to sink below the threshold within a very short period of time. Similar behavior can be observed when the node starts to operate in ES-mode shortly after $E_{\text{residual}}(i)$ becomes smaller than $E_{\text{threshold}}(i)$. These repeated mode changes reduce system reliability and performance. Therefore, we use an energy window ψ which mitigates the effect of the repeated mode changes, as shown in Figure 3 and Algorithms 1 and 2.

4. Experimental Verification

To verify the performance of our proposed scheme, we measured the energy consumed while performing each encryption technique in a sensor node. After that, we applied the encryption techniques to a solar-powered WSN and measured the energy consumed in a node. In addition, we performed a simulation comparing our scheme with the WSNs that use only one type of encryption algorithm.

4.1. Energy Consumption Measurement of Encryption Algorithms. This work is the first proposal to use the combination of the symmetric and asymmetric encryptions according to the current available energy. Therefore, there is no existing related scheme. In order to evaluate the performance of the proposed scheme, we compared three cases, (1) only use of symmetric encryption, (2) use of symmetric and asymmetric combination, and (3) only use of asymmetric encryption, instead of comparing the proposed scheme with others. Since the sensor nodes have the constraints of energy and computation, the adoptable types of cryptosystems, especially in the case of asymmetric cryptosystems, are very limited. Although there have been some attempts [28–30] to use the asymmetric encryption, Elliptic Curve Cryptography (ECC) is considered as the most feasible choice among them due to its fast computation, small size of keys, and compact signature. For instance, the ECC scheme needs only 160 bits to provide the same level of security as 1024-bit RSA. The well-known ECC schemes are the Elliptic Curve Diffie-Hellman (ECDH) key agreement scheme, the Elliptic Curve Digital Signature Algorithm (ECDSA), and the Elliptic Curve Integrated Encryption Scheme (ECIES). Among them, ECIES is used as a public-key encryption scheme in this work.

Table 4: Experimental parameters.

Parameter	Description
Sensor node	TelosB
Transmit power	0 dbm
MAC	LPL (based on B-MAC)
Data size (included timestamps and node IDs)	60 B
Key size of AES-128	128 bits
Key size of ECIES	256 bits

Table 5: Energy consumption of each cryptosystem algorithm.

Algorithm	Action	Energy
AES-128	Encryption	0.078 mJ
	Decryption	0.19 mJ
ECIES	Encryption	96 mJ
	Decryption	48 mJ

First, we measured the energy consumed while executing AES-128 and ECIES at a sensor node. Detailed experimental parameters are described in Table 4. TelosB [31], a TinyOS-based sensor platform, was used with 0 dbm transmit power and LPL (Low Power Listening) MAC protocol, which is a variation of B-MAC. Table 5 lists the results of measuring the energy consumed in the encryption and decryption of each algorithm. It shows that ECIES consumed 1,230 times and 250 times more energy than AES-128 during encryption and decryption, respectively.

Second, when nodes in ER-mode or ES-mode transmitted data, the energy consumption of the intermediate nodes (excluding the sink node and the node that generated the data first) was measured by applying the above two algorithms and is summarized in Table 6. When a node received a packet that was encrypted by AES-128, the node consumed 0.28 mJ if it was in ES-mode and consumed 96.19 mJ if it was in ER-mode. This is because the node in ER-mode should have decrypted the received packet by using AES-128 and then reencrypted it by using ECIES, while the node in ES-mode decrypted and reencrypted the received packet by using only AES-128. Meanwhile, if the received packet was encrypted by using ECIES, the node consumed only 0.054 mJ regardless of its mode, since the node should have only relayed the packet without any decrypting or reencrypting. Recall that the packet once encrypted by ECIES did not need to be decrypted or reencrypted.

Table 6: Energy consumption of cryptosystem in intermediate nodes.

Type of received packet	Mode of an intermediate node	Energy
Data encrypted by AES-128	ES-mode	0.285984 mJ
Data encrypted by ECIES		0.054009 mJ
Data encrypted by AES-128	ER-mode	96.19254 mJ
Data encrypted by ECIES		0.054009 mJ

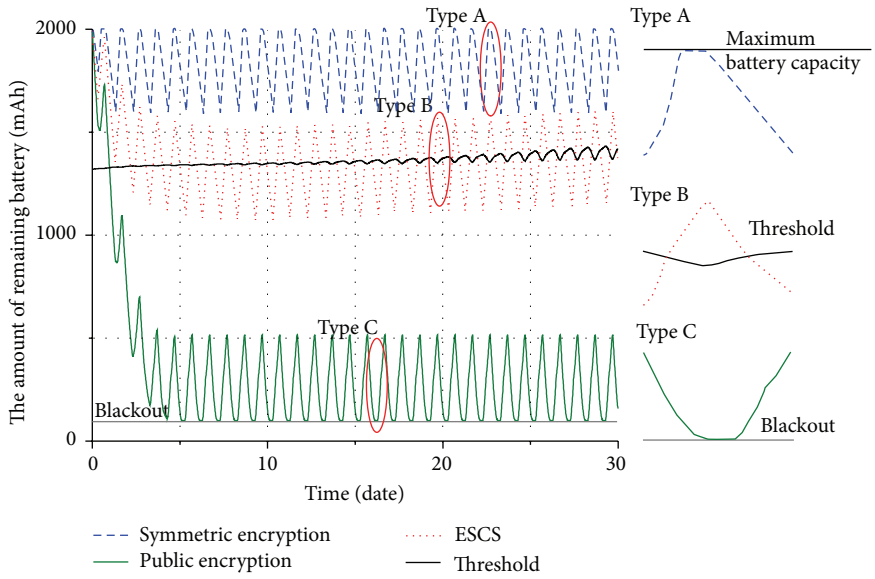

Figure 4: The comparison of the residual energy and threshold.

In our ESCS only in the case of a solar-powered node having enough residual energy (which meant it was operating in ER-mode) did it try to apply ECIES to the received packet. This effort resulted in increasing the level of security level as well as decreasing the energy consumption of the intermediate nodes.

4.2. Network-Wide Simulation. A simulation was performed to compare the performance of our ESCS scheme with that of other techniques in sensor networks. For this simulation, a network that consisted of 20 solar-powered nodes and an energy-adaptive location-based routing technique [32] was used. The important parameters are listed in Table 7.

4.2.1. Energy Consumption. To analyze the performance of ESCS in the aspect of energy consumption, the traces of residual energy and blackout time are compared in Figures 4 and 5, respectively. This was measured in each network which used only the symmetric-key, public-key, and ESCS schemes, respectively. As shown in these figures, in the network using only the public-key scheme, the amount of remaining energy in a node decreased fast and finally blackout occurred on about 4th day. This is because the node consumed energy faster than the rate of energy harvesting.

On the other hand, in the case of using only the symmetric-key scheme, a node consumed much less energy than the harvesting energy; thus a blackout never happened. However, most of the harvested energy could be wasted without being utilized, due to the limitation of battery capacity.

Unlike these two cases, our ESCS controls the energy consumption rate dynamically by selecting the security scheme adaptively depending on the amount of residual energy. Therefore, there is not only no blackout but also no waste of energy in a network using ESCS.

Table 7: Simulation environments.

Parameter	Description
Number of nodes	20
Battery capacity	2000 mAh
Transmission power	0 dbm
Transmission range	10 m
Size of network area	1000 m^2
Node deployment	Random
Duty circle	10%
Weather	Random
Simulation time	30 days
Routing	Energy-aware location-based routing
MAC	B-MAC
Simulation tool	SolarCastalia [36]

4.2.2. Level of Data Security. To analyze how much our ESCS contributes to the enhancement of the data security,

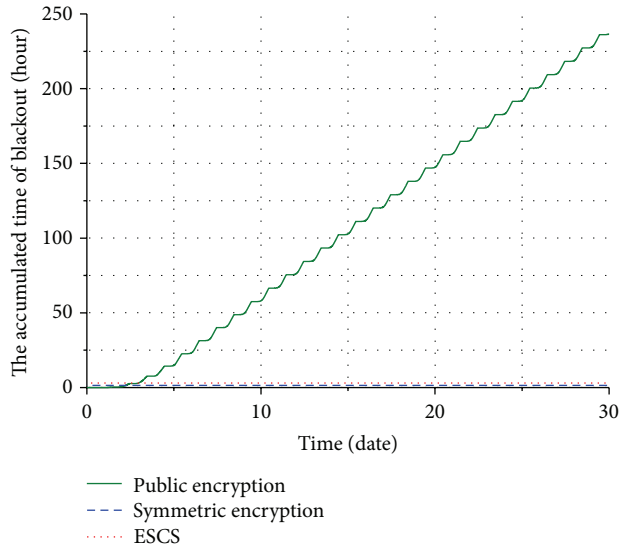

Figure 5: The comparison of the blackout time.

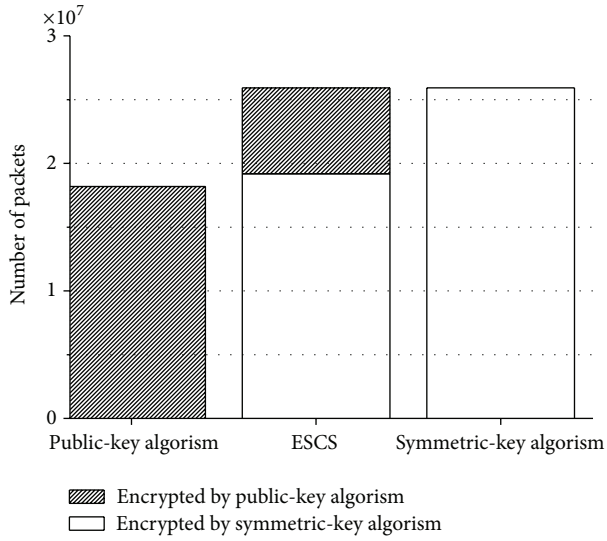

Figure 6: The number of packets encrypted by a public-key scheme in the sink.

we measured the number of packets which are successfully delivered to a sink node and also the number of those packets which were encrypted by a public-key scheme.

As shown in Figure 6, in the WSN where only a symmetric-key scheme was applied, the sink node could gather 135% more packets than in the case where only a public-key scheme was applied. This is because the node in WSN, where a public-key scheme was applied, was likely to be blacked out. However, note that perhaps the WSN which used only a symmetric-key scheme could reap more packets; the level of data security was inevitably lower. In the WSN to which ESCS was applied, the packet loss was almost zero just as in the case where only a symmetric-key scheme was applied, and 26% of packets received in the sink node were encrypted by a public-key algorithm. From these results, we confirm that our ESCS can utilize the harvested energy efficiently in order to enhance the level of data security.

5. Conclusion

Unlike battery-powered WSNs in which the goal is to extend network lifetime by reducing energy consumption, harvested energy should be fully utilized as long as nodes operate permanently in solar-powered WSNs. In this paper, we propose an energy-aware security level control technique that increases encryption level and energy efficiency by sufficiently utilizing the harvested energy. In this scheme, nodes are classified into ES-mode and ER-mode according to their remaining energy level; then, data is transmitted using a symmetric-key method with low-energy consumption and a low encryption level, as well as with a public-key method with high-energy consumption and a high encryption level. The simulation verified that the proposed technique increased both the encryption level and energy efficiency by utilizing the remaining energy, compared to other techniques.

Competing Interests

The authors declare no conflict of interests.

Acknowledgments

This work was supported by the Soongsil University Research Fund of 2013.

References

[1] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[2] J. Kim, H. Lee, J. Yi, M. Park, and D. Noh, "Energy-aware data encryption for solar-powered wireless sensor networks," in *Proceedings of the IEEE Asia Pacific Wireless Communications Symposium (IEEE VTS APWCS '15)*, Singapore, 2015.

[3] J. M. Gilbert and F. Balouchi, "Comparison of energy harvesting systems for wireless sensor networks," *International Journal of Automation and Computing*, vol. 5, no. 4, pp. 334–347, 2008.

[4] A. H. Sellers and P. J. Robinson, *Contemporary Climatology*, Longman Scientific & Technical, New York, NY, USA, 1986.

[5] J. L. Monteith and M. H. Unsworth, *Principles of Environmental Physics*, Edward Arnold, London, UK, 1990.

[6] T. Liu, C. M. Sadler, P. Zhang, and M. Martonosi, "Implementing software on resource-constrained mobile sensors: experiences with impala and ZebraNet," in *Proceedings of the 2nd Annual Mobile Systems, Applications, and Services (MobiSys '04)*, pp. 256–269, ACM, Boston, Mass, USA, June 2004.

[7] G. Tolle, J. Polastre, R. Szewczyk et al., "A macroscope in the redwoods," in *Proceedings of the 3rd Annual ACM International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. 51–63, San Diego, Calif, USA, November 2005.

[8] C. M. Vigorito, D. Ganesan, and A. G. Barto, "Adaptive control of duty cycling in energy-harvesting wireless sensor networks," in *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '07)*, pp. 21–30, IEEE, San Diego, Calif, USA, June 2007.

[9] C. Alippi and C. Galperti, "An adaptive system for optimal solar energy harvesting in wireless sensor network nodes," *IEEE Transactions on Circuits and Systems. I. Regular Papers*, vol. 55, no. 6, pp. 1742–1750, 2008.

[10] J. Taneja, J. Jeong, and D. Culler, "Design, modeling and capacity planning for micro-solar power sensor networks," in *Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 407–418, St. Louis, Mo, USA, April 2008.

[11] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 463–468, IEEE, New York, NY, USA, April 2005.

[12] S. Round, B. P. Otis, Y. Chee, J. M. Rabaey, and P. Wright, "A 1.9 GHz RF transmit beacon using environmentally scavenged energy luminance scaling," in *Proceedings of the International Symposium on Low Power Electronics and Design*, Seoul, Republic of Korea, August 2003.

[13] T. Voigt, H. Ritter, and J. Schiller, "Utilizing solar power in wireless sensor networks," in *Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks (LCN '03)*, pp. 416–422, Bonn, Germany, 2003.

[14] A. Kansal, J. Hsu, M. Srivastava, and V. Raghunathan, "Harvesting aware power management for sensor networks," in *Proceedings of the 43rd Annual Design Automation Conference*, pp. 651–656, San Francisco, Calif, USA, 2006.

[15] D. Noh, D. Lee, and H. Shin, "QoS-aware geographic routing for solar-powered wireless sensor networks," *IEICE Transactions on Communications*, vol. 90, no. 12, pp. 3373–3382, 2007.

[16] D. Noh, I. Yoon, and H. Shin, "Low-latency geographic routing for asynchronous energy-harvesting WSNs," *Journal of Networks*, vol. 3, no. 1, pp. 78–85, 2008.

[17] D. K. Noh and J. Hur, "Using a dynamic backbone for efficient data delivery in solar-powered WSNs," *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1277–1284, 2012.

[18] Y. Yang, L. Wang, D. K. Noh, H. K. Le, and T. F. Abdelzaher, "SolarStore: enhancing data reliability in solar-powered storage-centric sensor networks," in *Proceedings of the 7th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys '09)*, pp. 333–346, ACM, Kraków, Poland, June 2009.

[19] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[20] P. Papadimitratos and Z. J. Haas, "Secure routing for mobile ad hoc networks," in *Proceedings of the SCS Communication Networks and Distributed System Modeling and Simulation Conference*, San Antonio, Tex, USA, January 2002.

[21] L. Hu and D. Evans, "Secure aggregation for wireless networks," in *Proceedings of the Symposium on Applications and the Internet Workshops (SAINT '03 Workshops)*, pp. 384–391, Orlando, Fla, USA, 2003.

[22] F. Ye, H. Luo, S. Lu, and L. Zhang, "Statistical En-route filtering of injected false data in sensor networks," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM '04)*, pp. 2446–2457, Hong Kong, March 2004.

[23] T. Kaya, G. Lin, G. Noubir, and A. Yilmaz, "Secure multicast groups on adhoc networks," in *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '03)*, pp. 94–102, 2003.

[24] NIST Standards, "Advanced encryption standard (AES)," FIPS PUB 197, 2001.

[25] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the 7th IEEE International Conference on Information Processing in Sensor Networks (IPSN '08)*, pp. 245–256, St. Louis, Mo, USA, April 2008.

[26] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 197–215, IEEE, Washington, DC, USA, May 2003.

[27] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 539–550, 1999.

[28] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs," in *Cryptographic Hardware and Embedded Systems—CHES 2004*, pp. 119–132, Springer, Berlin, Germany, 2004.

[29] D. J. Malan, M. Welsh, and M. D. Smith, "A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography," in *Proceedings of the First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (IEEE SECON '04)*, IEEE, 2004.

[30] H. Wang and Q. Li, "Efficient implementation of public key cryptosystems on mote sensors (short paper)," in *Information and Communications Security*, vol. 4307 of *Lecture Notes in Computer Science*, pp. 519–528, Springer, Berlin, Germany, 2006.

[31] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 364–369, Seattle, Wash, USA, April 2005.

[32] R. Wu, M. Chen, Y. Su, and H. J. Siddiqui, "A novel location-based routing algorithm for energy balance in wireless sensor networks," in *Proceedings of the WRI International Conference on Communications and Mobile Computing (CMC '09)*, pp. 568–572, Yunnan, China, January 2009.

[33] Crossbow Products, http://www.openautomation.net/uploadsproductos/micaz_datasheet.pdf.

[34] Crossbow Products, http://wsn.cse.wustl.edu/images/e/e3/Imote2_Datasheet.pdf.

[35] Jennic Ltd, "JN5139 Wireless Microcontroller (IEEE 802.15.4 and ZigBee)," 2008, http://www.glynstore.com/content/docs/jennic/JN-DS-JN5139-001-1v9.pdf.

[36] J. M. Yi, M. J. Kang, and D. K. Noh, "SolarCastalia: solar energy harvesting wireless sensor network simulator," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 415174, 10 pages, 2015.