Wireless Netw
DOI 10.1007/s11276-015-0938-8

Location-based data encryption for wireless sensor network using dynamic keys

Han-Yu Lin[1]

Abstract Secure data transmission for the wireless sensor network (WSN) is always an important issue. The technique of traditional authenticated encryption allows a sensor node to generate a ciphertext which can only be decrypted and authenticated by a designated data aggregator. The convertible property further enables the aggregator to announce an ordinary signature for public verification. To alleviate the harm of key exposure, dynamic key systems are especially suitable for implementing in the large-scale deployment environments such as WSNs. Combining the concept of location and the merits of dynamic keys, we propose a location-based data encryption scheme for WSNs. To the best of our knowledge, this is the first concrete construction considering the properties of location and dynamic keys in WSNs. The proposed scheme not only is conversion-free, but also provides unlimited time periods and random-access key-updates. Moreover, we utilize some reduction models to prove the security of our protocol.

Keywords WSN · Location · Key-insulation · Authenticated encryption · Message linkages

1 Introduction

As wireless networks [12, 14] develop over time, more and more wireless applications can be found out in our daily life. To guarantee the data transmission and communication safety, the security requirements of integrity, authentication, confidentiality and non-repudiation [5, 18, 23] must be fulfilled. One of the popular wireless applications is WSNs [3, 4, 16] which consist of a large scale of distributed devices with embedded sensors. These sensors can monitor lots of environmental conditions such as sound, pressure and temperature, etc., and hence are often utilized in military surveillance. Up to the present, the WSN can be also seen in many non-military fields like home automation, healthcare systems, traffic control, fire detection and so on.

Because each sensor node has a microprocessor and a wireless communication device, it could be regarded as a mini computer with limited computing power and storage. Energized by batteries, these sensor nodes are usually deployed in wilderness areas without replacement. For facilitating the data transmission between a data aggregator and its nearby sensor nodes, each sensor has a preinstalled secret key. However, once the key is compromised by some malicious adversary, it will endanger the communication messages delivered to the data aggregator. It thus can be seen that a proper data encryption mechanism with efficient key update procedure for WSNs is crucial for increasing the security strength and solving the node-compromised problem.

1.1 Related works

In 1984, Shamir [22] introduced the famous ID-based system in which a private key generation center (PKG) is responsible for generating every user's private key with a trapdoor

✉ Han-Yu Lin
lin.hanyu@msa.hinet.net

[1] Department of Computer Science and Engineering, National Taiwan Ocean University, 2, Beining Road, 202 Keelung, Taiwan, ROC

Published online: 05 April 2015

one-way function. Without the secret, no one can perform the trapdoor one-way function, so as to guarantee the confidentiality of private keys. The corresponding public key is explicitly verified, as it consists of some identity information (such as the user name and e-mail address, etc.) To make the ID-based system suitable for practical implementation, in 2001, Boneh and Franklin [2] proposed an ID-based system from the Weil pairing. Since then, ID-based systems have been widely adopted for designing various security protocols. However, once a user's private key is accidentally compromised, the corresponding confidential information might be decoded by a malicious adversary.

To withstand above key exposure attacks, Dodis et al. [6, 7] addressed the first key-insulated system with dynamic private keys. In a key-insulated system, each user owns two private keys. One is a short-term private key kept secret by the user and the other is a long-term one stored in a physically-secure but computation limited device (called base or helper). With the assistance of the helper, a user can periodically update his short-term private key for performing various security protocols such as public key encryption and digital signature schemes [8, 20, 21] at different time periods. Note that the public key of each user is still unchanged. The general idea of key-insulated systems is that even if an adversary has the knowledge of all previous private keys for some user, he cannot perform any private key operation on behalf of the user in relation to the current time period.

Considering the advantages of ID-based systems and the key-insulated ones, Hanaoka et al. [10] proposed the first ID-based key-insulated encryption (KIE). In their literature, they demonstrated how to construct a partially collusion resistant hierarchical identity-based encryption from arbitrary IBE. The next year, Zhou et al. [28] proposed an identity-based key-insulated signature scheme based on the computational Diffie–Hellman problem (CDHP). To reduce the possibility of helper exposure, Hanaoka et al. [9] adopted two independent helpers to construct a KIE scheme. The two helpers will be alternatively selected for helping with user's key-updates. Such a scheme is called the parallel-KIE. It can be seen that a parallel-KIE reaches a higher security level with respect to the helper's security. So far, lots of protocols [11, 17, 24, 26] for key-insulated systems have been proposed.

In 2005, Lazos et al. [15] utilized the property of location to address a location-based mechanism for securing the wireless security. The next year, Zhang et al. [27] further proposed a location-based compromise-tolerant security protocol for WSNs. In their scheme, the private key of each node is bound with its ID and geographic location. They also developed a so-called neighborhood authentication scheme to reduce the impact of compromised nodes. Nevertheless, both Lazos et al. and Zhang et al. failed to incorporate the advantage of dynamic keys into the design of their security protocols.

1.2 Our contribution

Consider the security in WSNs, in this paper, we combine the concept of location with the advantages of dynamic keys to construct a concrete location-based data encryption mechanism for WSNs. Unlike previous location-based security mechanisms which only used unique private keys for sensors in WSNs, we incorporate the dynamic property of key-insulated systems into the design and construction of the proposed scheme. Without redeployment, each sensor node can periodically update its private key at different time periods. The proposed scheme possesses the properties of conversion-free, unlimited time periods and random-access key-updates. Moreover, a theoretic proof model is adopted to demonstrate the feasibility of our work.

2 Proposed protocol

There are four involved parties of our protocol, including a PKG, a helper (embedded chipset), a sensor node and a designated data aggregator. Initially, the PKG is responsible for generating each node's initial private key and a master helper key. At the beginning of every time period i, a sensor node can update its i-th short-term private key with the helper's aid. Let (ID_A, l_A) and (ID_B, l_B) be a sensor node and a data aggregator along with their corresponding locations, respectively. Figure 1 shows how the four parties collaborate with each other. Details of each phase are described below:

- **Setup:** Taking as input 1^k where k is a security parameter, the PKG first chooses a master secret key $s \in_R Z_q$ and a master helper key $w \in_R Z_q$. The corresponding public keys are computed as $P_{TA} = sP$ and $P_{HK} = wP$. The master helper key w is then prestored in the helper. Let $(\boldsymbol{G}_1, +)$ and $(\boldsymbol{G}_2, \times)$ be two groups of the same prime order q where $|q| = k$, P a generator of order q over $\boldsymbol{G}_1$, e: $\boldsymbol{G}_1 \times \boldsymbol{G}_1 \rightarrow \boldsymbol{G}_2$ a bilinear pairing, H: $\{0, 1\}^k \rightarrow \boldsymbol{G}_1$, F_1: $\boldsymbol{G}_1 \times \boldsymbol{G}_2 \rightarrow Z_q$, F_2: $\{0, 1\}^k \times \{0, 1\}^* \times \boldsymbol{G}_1 \times \boldsymbol{G}_2 \times \boldsymbol{G}_2 \rightarrow Z_q$ and F_3:

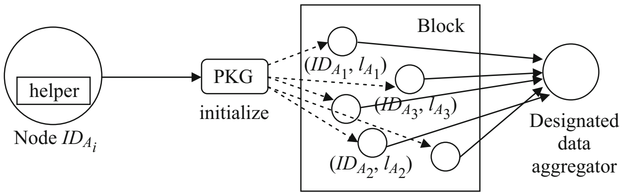

Fig. 1 A block diagram illustrating the four collaborated parties of our protocol

$Z_q \to Z_q$ collision resistant hash functions. The public parameter *params* includes $\{P_{TA}, P_{HK}, \mathbf{G}_1, \mathbf{G}_2, q, P, e, H, F_1, F_2, F_3\}$.

- **KeyExtract (KE):** The PKG computes the initial private key for ID_A as $S_{A,\,0} = sH(ID_A, l_A) + wH(ID_A, l_A, 0)$. The corresponding public key is computed as $\sigma_A = e(P_{TA}, H(ID_A, l_A))$.
- **KeyUpdate (KU):** For any time period $i \in \{1, \ldots, N\}$, the helper first generates the corresponding helper key as $HK_{A,\,i} = w[H(ID_A, l_A, i) - H(ID_A, l_A, i - 1)]$. Then the sensor node can update its private key by computing $S_{A,\,i} = S_{A,\,i-1} + HK_{A,\,i}$. The values $(S_{A,\,i-1}, HK_{A,\,i})$ are deleted subsequently. Figure 2 illustrates the update of private keys with different time periods.
- **Encryption (EN):** For encrypting a packet $M = M_1 \parallel M_2 \parallel \ldots \parallel M_l$ at any time period $i \in \{1, \ldots, N\}$, ID_A chooses $r \in_R Z_q$ and $C_0 = 0$ to compute $R = rP$, $d_B = [e(P_{HK}, H(ID_B, l_B, i))\sigma_B]^r$, $d_A = [e(P_{HK}, H(ID_A, l_A, i))\sigma_A]^r$, $C_v = M_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B))$, for $v = 1, 2, \ldots, l$, and $Q = (r + F_2(i, M, R, d_A, d_B))S_{A,\,i}$. The ciphertext for M is $\delta = (i, C_1, C_2, \ldots, C_l, R, Q, d_A)$.
- **Decryption-and-Verification (DV):** To decrypt δ, the data aggregator ID_B first derives $d_B = e(R, S_{B,\,i})$, computes $M_v = C_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B))^{-1}$, for $v = 1, 2, \ldots, l$, and then recovers the original packet $M = M_1 \parallel M_2 \parallel \ldots \parallel M_l$. ID_B further verifies its authenticity by checking if $e(P,Q) = d_A \cdot [\sigma_A \cdot e(P_{HK}, H(ID_A, l_A, i))]^{F_2(i,M,R,d_A,d_B)}$. Figure 3 shows the EN and DV processes for a packet $M = M_1 \parallel M_2 \parallel \ldots \parallel M_l$.

The correctness of each datagram M_v is shown as follows:

$$\begin{aligned}
&C_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, e(R, S_{B,i})))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, e(R, sH(ID_B, l_B) \\
&\qquad + wH(ID_B, l_B, i))))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, e(R, sH(ID_B, l_B))e \\
&\qquad (R, wH(ID_B, l_B, i))))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, e(sR, H(ID_B, l_B))e \\
&\qquad (wR, H(ID_B, l_B, i))))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, e(srP, H(ID_B, l_B))e \\
&\qquad (wrP, H(ID_B, l_B, i))))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, e(P_{TA}, rH(ID_B, l_B))e \\
&\qquad (P_{HK}, rH(ID_B, l_B, i))))^{-1} \\
&\quad = C_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B))^{-1} \\
&\quad = M_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B)) \cdot F_3(C_{v-1} \oplus F_1(R, d_B))^{-1} \\
&\quad = M_v
\end{aligned}$$

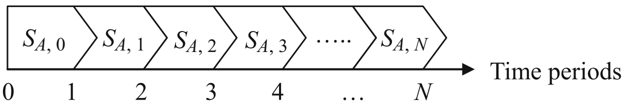

Fig. 2 The update of private keys with different time periods

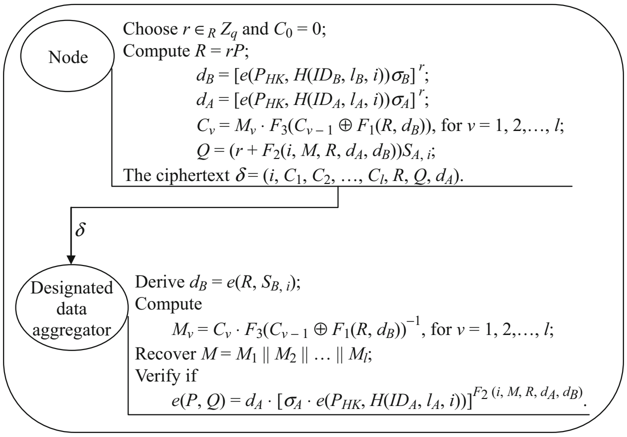
Node
Choose $r \in_R Z_q$ and $C_0 = 0$;
Compute $R = rP$;
$d_B = [e(P_{HK}, H(ID_B, l_B, i))\sigma_B]^r$;
$d_A = [e(P_{HK}, H(ID_A, l_A, i))\sigma_A]^r$;
$C_v = M_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B))$, for $v = 1, 2, \ldots, l$;
$Q = (r + F_2(i, M, R, d_A, d_B))S_{A,\,i}$;
The ciphertext $\delta = (i, C_1, C_2, \ldots, C_l, R, Q, d_A)$.
δ
Designated data aggregator
Derive $d_B = e(R, S_{B,\,i})$;
Compute
$M_v = C_v \cdot F_3(C_{v-1} \oplus F_1(R, d_B))^{-1}$, for $v = 1, 2, \ldots, l$;
Recover $M = M_1 \parallel M_2 \parallel \ldots \parallel M_l$;
Verify if
$e(P, Q) = d_A \cdot [\sigma_A \cdot e(P_{HK}, H(ID_A, l_A, i))]^{F_2(i, M, R, d_A, d_B)}$.

Fig. 3 The EN and DV processes for a packet M

The correctness for the authenticity of the recovered packet M can also be verified below:

$$\begin{aligned}
&e(P,Q) \\
&\quad = e(P, (r + F_2(i,M,R,d_A,d_B))S_{A,i}) \\
&\quad = e(P, (r + F_2(i,M,R,d_A,d_B))(sH(ID_A,l_A) + wH(ID_A,l_A,i))) \\
&\quad = e(P, (r + F_2(i,M,R,d_A,d_B))(sH(ID_A,l_A))) \\
&\quad \times e(P, (r + F_2(i,M,R,d_A,d_B))(wH(ID_A,l_A,i))) \\
&\quad = e(P, rsH(ID_A,l_A))e(P, F_2(i,M,R,d_A,d_B)(sH(ID_A,l_A))) \\
&\quad \times e(P, rwH(ID_A,l_A,i))e(P, F_2(i,M,R,d_A,d_B)(wH(ID_A,l_A,i))) \\
&\quad = \sigma_A^r \times \sigma_A^{F_2(i,M,R,d_A,d_B)} e(P_{HK}, rH(ID_A,l_A,i)) \\
&\quad \times e(P_{HK}, F_2(i,M,R,d_A,d_B)H(ID_A,l_A,i)) \\
&\quad = d_A \times \sigma_A^{F_2(i,M,R,d_A,d_B)} e(P_{HK}, F_2(i,M,R,d_A,d_B)H(ID_A,l_A,i)) \\
&\quad = d_A \times [\sigma_A \times e(P_{HK}, H(ID_A,l_A,i))]^{F_2(i,M,R,d_A,d_B)}
\end{aligned}$$

3 Security analyses

Two fundamental security assumptions employed in our protocol are CDHP over elliptic curves and BDHP described below [2]:

CDHP (Computational Diffie–Hellman Problem) over elliptic curves Let P be a random point in $\mathbf{G}_1$ and $a, b \in_R Z_q^*$. It is computationally infeasible to derive abP from a given instance (P, aP, bP)

BDHP (Bilinear Diffie–Hellman Problem) Let P be a generator, $A = aP$, $B = bP$ and $C = cP$ for some a, b, $c \in Z_q$. It is computationally infeasible to derive $e(P, P)^{abc} \in \boldsymbol{G}_2$ from a given an instance $(P, A, B, C) \in G_1^4$.

When it comes to the security requirements of any hybrid security protocol providing encryption and digital signature, we should consider the property of confidentiality against indistinguishability under adaptive chosen-ciphertext attacks (IND-CCA2) and that of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA). We use well-defined theoretic proof models to show the security of our approach as Theorems 1 and 2.

Theorem 1 *The proposed protocol is* (t, ε)*-secure against IND-CCA2 if no probabilistic adversary is able to break the BDHP within polynomial-time and with the advantage* ε'.

Proof In the notion of adaptive chosen-ciphertext attacks, we assume that within the polynomial-time t, there is a probabilistic adversary $\mathcal{A}$ who is able to break the proposed scheme with non-negligible advantage ε. The adversary $\mathcal{A}$ is permitted to issue the queries of H, F_1, F_2, KE, HK (helper key), KU, EN and DV, meaning that the result for these queries can only be obtained through a challenger. The total times limitations for above queries are q_H, q_{F_1}, q_{F_2}, q_{KE}, q_{HK}, q_{KU}, q_{EN} and q_{DV}, respectively. We claim that if $\mathcal{A}$ exists, a polynomial-time algorithm $\mathcal{B}$ extended from $\mathcal{A}$ is able to compute $e(P, P)^{xyz}$ for the BDHP instance (P, xP, yP, zP). Let F_3 be a collision resistant hash function. In the following processes, $\mathcal{B}$ acts as a challenger to $\mathcal{A}$.

Setup: $\mathcal{B}$ first initializes the Setup(1^k) phase to obtain $params = \{\boldsymbol{G}_1, \boldsymbol{G}_2, q, P, e, F_3\}$ and sets $P_{TA} = uP$, $P_{HK} = xP$ where $u \in_R Z_q$. Then $(params, P_{TA}, P_{HK})$ is sent to the adversary $\mathcal{A}$.

Phase 1: For each query made by $\mathcal{A}$ in the time period $i \in \{1, \ldots, N\}$, $\mathcal{B}$ responses as follows:

- $H(ID_j, l_j)$ *queries:* $\mathcal{B}$ first checks a maintained table H_list for previous inquiries. If there is no matched entry, $\mathcal{B}$ chooses $h_j \in_R Z_q$, adds one entry $(ID_j, l_j, h_j, h_j P)$ and then returns $h_j P$.
- $F_1(R_j, d_j)$ *queries:* $\mathcal{B}$ first searches a maintained table F$_1$_list for previous inquiries. If there is no matched entry, $\mathcal{B}$ chooses $f_1 \in_R Z_q$, inserts one entry (R_j, d_j, f_1) and finally returns f_1.
- $F_2(i, M_j, R_j, d_j, d'_j)$ *queries:* $\mathcal{B}$ first checks a maintained table F$_2$_list for previous inquiries. If no entry matches, $\mathcal{B}$ chooses $f_2 \in_R Z_q$, adds one entry $(i, M_j, R_j, d_j, d'_j, f_2)$ and then returns f_2.
- $KE(ID_j)$ *queries:* $\mathcal{B}$ returns the initial private key $S_{j,\ 0} = h_j(uP) + (h_{j,\ 0})xP$.
- $HK(i, ID_j, l_j)$ *queries:* $\mathcal{B}$ returns the helper key $HK_{j,\ i} = (h_{j,\ i})xP - (h_{j,\ i\ -\ 1})xP$.
- $KU(i, ID_j, l_j)$ *queries:* $\mathcal{B}$ returns the private key $S_{j,\ i} = h_j(uP) + (h_{j,\ i})xP$.
- $EN(i, M, ID_A, ID_B)$ *queries:* $\mathcal{B}$ first derives the private key $S_{A,\ i} = h_A(uP) + (h_{A,\ i})xP$ and follows the steps of EN algorithm to return a corresponding ciphertext $\delta = (i, C, R, Q, d_A)$.
- $DV(\delta, ID_A, ID_B)$ *queries:* $\mathcal{B}$ first computes the private key $S_{B,\ i} = h_B(uP) + (h_{B,\ i})xP$ and then runs the DV algorithm. If the recovered packet is valid, returns $\{M, \Omega = (i, R, Q, d_A, d_B)\}$; else, an error symbol ¶ is returned.

Challenge: $\mathcal{A}$ chooses two fresh identities (ID_A^*, ID_B^*), two packets, M_0 and M_1, of equal length and a time period $i^* \in \{1, \ldots, N\}$. The challenger $\mathcal{B}$ flips a coin $\lambda \leftarrow \{0, 1\}$ and produces a ciphertext δ^* for $(i^*, M_\lambda, ID_A^*, ID_B^*)$ as follows:

1. Insert the entry $((ID_B^*, l_B{}^*, i^*), \text{null}, yP)$ into H_list, i.e., implicitly define $H(ID_B^*, l_B{}^*, i^*) = yP$ where y is unknown to $\mathcal{B}$.
2. Choose $Q^* \in_R \boldsymbol{G}_1$, $C_0^* = 0$ and set $R^* = zP$;
3. Compute $d_A^* = e(uP, (h_A^*)(zP)) \cdot\ e(xP, (h_{A,i^*}^*))zP)$;
4. Choose $f_1^* \in_R Z_q$ and insert the entry $(R^*, \text{null}, f_1^*)$ into F$_1$_list, i.e., implicitly define $F_1(R^*, d_B^*) = f_1^*$ where d_B^* is unknown to $\mathcal{B}$.
5. Compute $C_v^* = M_v^* \cdot F_3(C_{v-1}^* \oplus f_1^*)$, for $v = 1, 2, \ldots, l$.

The ciphertext $\delta^* = (i^*, C_1^*, C_2^*, \ldots, C_l^*, R^*, Q^*, d_A^*)$ is then sent to $\mathcal{A}$ as a target challenge.

Phase 2: $\mathcal{A}$ is able to issue new queries as those stated in Phase 1 except the HK(i^*, ID_B^*), KU(i^*, ID_B^*) and DV$(\delta^*, ID_A^*, ID_B^*)$ queries. $\mathcal{B}$ might directly terminate in the event that $\mathcal{A}$ queries HK$(i^* + 1, ID_B^*)$. When $\mathcal{A}$ asks DV(δ, ID_A, ID_B^*) where $\delta = (i^*, C_1, C_2, \ldots, C_l, R, Q, d_A)$, $\mathcal{B}$ checks F$_1$_list table for a possible entry (R_j, d_j, f_1) where $R_j = R$ and then computes $M_v = C_v \cdot F_3(C_{v\ -\ 1} \oplus f_1)^{-1}$, for $v = 1, 2, \ldots, l$. If $e(P, Q) = d_A \cdot [\sigma_A \cdot e(uP, H(ID_A, l_A, i))]^{F_2(i^*, M, R, d_A, d_B)}$, the packet M and its signature $\Omega = (i^*, R, Q, d_A, d_j)$ is returned. Otherwise, send an error symbol ¶ back.

Analysis of the game: In Phase 2, $\mathcal{B}$ can derive the private key $S_{A^*,\ i^*} = (h_A^*)(uP) + (h_{A,i^*}^*)xP$ where $i^* \in \{1, \ldots, N\}$ to respond each EN query in relation to ID_A^*. Consider the case that a DV(δ, ID_A, ID_B^*) query might return ¶ for a valid $\delta = (i^*, C_1, C_2, \ldots, C_l, R, Q, d_A)$ seeing that the corresponding $F_1(R, d_B^*)$ query has never been made. However, the probability for such an event is not greater than $(q_{DV})2^{-k}$ for total DV inquiries. We express it as DV_ERR and Pr[DV_ERR] $\leq (q_{SRV})2^{-k}$. Additionally, when $\mathcal{A}$ queries HK($i^* + 1$, ID_B^*, l_B), $\mathcal{B}$ aborts directly. We represent such an event as HK_ERR and Pr[HK_ERR] $\leq ((N-1)q_H)^{-1}$. Note that in the challenge phase, $\mathcal{B}$ generates a simulated ciphertext $\delta^* = (i^*, C_1^*, C_2^*, \ldots, C_l^*, R^*, Q^*, d_A^*)$ where $H(ID_B^*, l_B{}^*, i^*) = yP$ and $R^* = zP$, which implies d_B^* is implicitly defined as

$$\begin{aligned} d_B^* &= e(P_{TA}, zH(ID_B^*, l_B^*))e(P_{HK}, zH(ID_B^*, l_B^*, i)) \\ &= e(uP, h_B^*(zP))e(xP, z(yP)) \\ &= e(uP, h_B^*(zP))e(P, P)^{xyz}. \end{aligned}$$

Let NA denote that the simulation is perfect and such probability depends on the event that the $F_1(R^*, d_B^*)$ query is never made in Phase 2. We use QF_1^* to stand for that such an event indeed happens in Phase 2. When the entire simulation is perfect, $\mathcal{A}$ has no advantage in returning a correct λ due to the randomness of simulation, i.e., $\Pr[\lambda' = \lambda \mid \mathrm{NA}] = 1/2$. Based on the theorem of probability inequality, we can also have $|\Pr[\lambda' = \lambda] - 1/2| \leq (1/2)\Pr[\neg NA]$. By the initial assumption, $\mathcal{A}$ has non-negligible probability ε to break the proposed scheme. Consequently, we can derive

$$\begin{aligned} \varepsilon &= |\Pr[\lambda' = \lambda] - 1/2| \\ &\leq (1/2)\Pr[\neg\mathrm{NA}] \\ &\leq (1/2)(\Pr[\mathrm{QF}_1^*] + \Pr[\mathrm{DV\_ERR}] + \Pr[\mathrm{HK\_ERR}]) \end{aligned}$$

which means that

$$\begin{aligned} \Pr[\mathrm{QF}_1^*] &\geq 2\varepsilon - \Pr[\mathrm{SRV\_ERR}] - \Pr[\mathrm{HK\_ERR}] \\ &\geq 2\varepsilon - \frac{q_{DV}}{2^k} - \frac{1}{(N-1)q_H} \end{aligned}$$

and the value $d_B^* = e(uP, h_B^*(zP))e(P, P)^{xyz}$ would be stored in F_1_list on condition that QF_1^* occurs. Hence, we claim that $\mathcal{B}$ is able to solve the given BDHP instance by computing $e(uP, h_B^*(zP))^{-1}d_B^*$ and the success probability is

$$\varepsilon' \geq (q_{F1}^{-1})\left(2\varepsilon - \frac{q_{DV}}{2^k} - \frac{1}{(N-1)q_H}\right).$$

□

Theorem 2 *The proposed protocol is* (t, ε)*-secure against EF-CMA if no probabilistic adversary is able to break the CDHP within polynomial-time and with the advantage* ε'.

Proof In the notion of adaptive chosen-message attacks, we suppose that within the polynomial-time t, there is a probabilistic adversary $\mathcal{A}$ who is able to break the proposed scheme with non-negligible advantage ε. The adversary is allowed to make queries defined in Theorem 1 except DV. We claim that if $\mathcal{A}$ exists, a polynomial-time algorithm $\mathcal{B}$ extended from $\mathcal{A}$ is able to compute xyP for the CDH instance (P, xP, yP). We will utilize the Forking Lemma [19] to prove this theorem. Let F_3 also be a collision resistant hash function and $\mathcal{B}$ acts as a challenger to $\mathcal{A}$.

Setup: $\mathcal{B}$ first initializes the Setup(1^k) phase to obtain $params = \{\boldsymbol{G}_1, \boldsymbol{G}_2, q, P, e, F_3\}$ and selects a random tape θ (consisting of a long sequence of random bits). Next $\mathcal{B}$ sets $P_{TA} = uP$, $P_{HK} = xP$ where $u \in_R Z_q$ and simulates two runs of the proposed protocol $\mathcal{A}$ on input ($params$, P_{TA}, P_{HK}, θ).

Phase 1: $\mathcal{A}$ can adaptively ask allowed queries and $\mathcal{B}$ responds just like those defined in Theorem 1. Note that when A queries $H(ID_A, l_A, i)$, $\mathcal{B}$ responses with yP. In the event that $\mathcal{A}$ queries HK($i + 1$, ID_A, l_A), $\mathcal{B}$ aborts directly.

Analysis of the game: Since $\mathcal{B}$ directly aborts whenever $\mathcal{A}$ asks HK($i + 1$, ID_A, l_A), we denote the situation as HK_ERR and Pr[HK_ERR] $\leq ((N-1)q_H)^{-1}$. Let EN-V be the case that a ciphertext $\delta^* = (i^*, C_1^*, C_2^*, \ldots, C_l^*, R^*, Q^*, d_A^*)$ for M^* in relation to (ID_A^*, ID_B^*) is valid, meaning that Pr[EN − V] = ε by the initial assumption. Consider the case that $\mathcal{A}$ produces a valid δ^* without querying $F_2(i^*, M^*, R^*, d_A^*, d_B^*)$. That is, $\mathcal{A}$ guesses the correct f_2, denoted as NH and we know that Pr[NH] $\leq 2^{-k}$. Then, we can express the probability that a forgery $\delta^* = (i^*, C_1^*, C_2^*, \ldots, C_l^*, R^*, Q^*, d_A^*)$ is valid and the corresponding F_2 query is also made as $\Pr[(\mathrm{EN} - \mathrm{V} \wedge \neg\mathrm{NH}) \mid \neg\mathrm{HK\_ERR}] \geq \left(\varepsilon - \frac{1}{2^k}\right)\left(1 - \frac{1}{(N-1)q_H}\right)$. When $i^* = i$ and $ID_A^* = ID_A$, we obtain

$$\begin{aligned} \varepsilon^* &= Pr[(\mathrm{EN} - \mathrm{V} \wedge \neg\mathrm{NH}) | \neg\mathrm{HK\_ERR} \\ &\quad \wedge (i^* = i, ID_A^* = ID_A)] \\ &\geq \left(\varepsilon - \frac{1}{2^k}\right)\left(1 - \frac{1}{(N-1)q_H}\right)\left(\frac{1}{N \cdot q_H}\right) \\ &= \left(\varepsilon - \frac{1}{2^k}\right)\left(\frac{(N-1)q_H - 1}{(N^2 - N)q_H^2}\right) \end{aligned}$$

$\mathcal{B}$ again runs the second simulation on the same input ($params$, P_{TA}, P_{HK}, θ) and responses with identical values as those in the first run. When $\mathcal{A}$ asks $F_2(i^*, M^*, R^*, d_A^*, d_B^*)$, $\mathcal{B}$ returns a different value $f_2{}^{**} \in_R Z_q$. Based on the "Forking lemma", if $\mathcal{A}$ finally generate another valid forgery $\delta^{**} = (i^*, C_1^*, C_2^*, \ldots, C_l^*, R^*, Q^{**}, d_A^*)$ with $i^* = i$, $ID_A^* = ID_A$ and $F_2(i^*, M^*, R^*, d_A^*, d_B^*) \neq F_2'(i^*, M^*, R^*, d_A^*, d_B^*)$, $\mathcal{B}$ could derive $e(P, Q^* - Q^{**}) = [e(uP, (h_{A^*})P) \cdot e(xP, yP)]^{(f_{2^*} - f_{2^{**}})}$ and then solve the given CDH instance by computing $xyP = (f_{2^*} - f_{2^{**}})^{-1}[(Q^* - Q^{**}) - (f_{2^*} - f_{2^{**}}) u(h_{A^*})P]$. To evaluate $\mathcal{B}$'s success probability, we can first utilize the "Splitting lemma" [19] to learn that the probability for $\mathcal{A}$ to generate a valid forgery in the second run is at least $(2^{-1}\varepsilon)^2 = 4^{-1}\varepsilon^2$. Since we have known that the probability to eventually create another valid $\delta^{**} = (i^*, C_1^*, C_2^*, \ldots, C_l^*, R^*, Q^{**}, d_A^*)$ with $F_2(i^*, M^*, R^*, d_A^*, d_B^*) \neq F_2{}'(i^*, M^*, R^*, d_A^*, d_B^*)$ by $\mathcal{A}$ is $q_{F_2}^{-1}$, the success probability for $\mathcal{B}$ after two simulation runs can be represented as $\varepsilon' \geq (4q_{F_2})^{-1}\left[\left(\varepsilon - \frac{1}{2^k}\right)\left(\frac{(N-1)q_H-1}{(N^2-N)q_H^2}\right)\right]^3$. □

Based on our location-based data encryption mechanism, we further consider the following practical attacks in WSNs from the perspective of message confidentiality. It should be noted that the proposed scheme is mainly designed for guaranteeing the end-to-end data transmission security, rather than an overall security solution to WSNs. For defeating some specific attacks such as the sinkhole attack which aims at causing the failure of network routing functions, we have to combine additional detecting or authentication protocol [27] to gain a more comprehensive protection in WSNs.

1. *Node Spoofing Attacks:* An adversary might attempt to impersonate any legitimate node or the designated data aggregator for obtaining confidential messages. However, all encrypted packets can only be decrypted with the private key of designated data node/aggregator. Based on the proofs of Theorem 2, we conclude that the adversary cannot derive the confidential information without the knowledge of valid private key.
2. *Sybil Attacks:* When launching this attack, an adversary might utilize forged identities to masquerade as a large number of nodes. In our scheme, if these nodes forward bogus data to any legitimate node, it will be detected during the DV phase, since bogus data cannot pass the verification procedure.
3. *Identity Replication Attacks:* In this attack, a compromised node will be replicated and put in different locations to cause the inconsistence of network routing information. Nevertheless, in our scheme, all forwarded routing messages must contain valid signature information (Q, d_A) to be verified and accepted by any legitimate node or the data aggregator. Therefore, the identity replication attack cannot work in the proposed scheme.
4. *Wormhole Attacks:* In this attack, an adversary first generates a wormhole link between different network locations and then uses this link to tunnel routing messages from one to the other location, so as to cause the chaotic situation. Yet, in our scheme, all transmitted messages can only be accepted if they contain valid signatures with respect to the sender's location and key information. Thus, our scheme is secure against the wormhole attack.
5. *Key Exposure Attacks:* In our scheme, each sensor node is equipped with a long-term private key w and a short-term session key $S_{A,\ i}$ which will be updated with different time periods. The former must be stored in a lightweight tamper-resistant on-chip security co-processor such as the microprocessors of ST19 and ST22 series ICs developed by STMicroelectronics Corporation. Xie et al. [25] also introduced a lightweight tamper-resistance design structure for WSNs to raise the secret-keeping capability for sensor nodes and withstand potential attacks. We note that the proposed scheme cannot prevent key exposure attacks entirely. Yet, in case that an adversary captures a compromised node, he can only obtain its short-term session key and then derive the related confidential messages for that specific time period. That is, the impact caused by the key exposure attacks can be mitigated and limited in only certain time periods depending on the exposed short-term session keys.

4 Performance evaluation

We make a performance evaluation with respect to the proposed scheme in this section. The evaluation is made in terms of computational costs and communication overheads. Since the data aggregator is usually more powerful than sensor nodes, we only take the capability of sensors into account. It is believed that the most time-consuming operation for pairing-based systems is the bilinear pairing computation. For simplicity, we will employ the number of required bilinear pairing to approximate the computational costs for sensor nodes. Let the symbol of 'T_B' be the time for performing one bilinear pairing. The detailed evaluation is demonstrated as Table 1.

In this table, we assume that a 32-bit Intel PXA255 processor at 400 MHz is adopted for sensor nodes. The PXA25x family is Intel's first generation of XScale processors and has been widely utilized in WSNs. Let the

Table 1 Performance evaluation for sensors

Processor	Intel PXA255 32-bit/400 MHz
Item	
Computation complexity for the EN process	$\approx 4\ T_B$
Energy consumption for the EN process	≈102 mJ
Communication costs for the EN process	$(l+4)\lvert q\rvert$

order q of $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ be a 160-bit prime and the bilinear map e be Tate pairing [1]. According to Zhang et al.'s analyses [27], a Tate pairing on the Intel PXA255 processor at 400 MHz roughly takes 62.04 ms and consumes 25.5 mJ, which helps derive that a sensor might approximately spend 248.16 ms and consume 102 mJ for running the EN process of the proposed scheme. As for the communication overhead, we evaluate it with the actual message length. One can observe that the transmitted messages include (i, C_1, C_2, …, C_l, R, Q, d_A). That is, the total communication overheads are $(l + 4)|q|$. Based on the Intel PXA255 processor specification [13], the energy consumption of 121 mW is required in idle mode while 411 mW power consumption is needed in active mode. The EN process of our protocol only takes roughly 102 mJ; therefore, we can conclude that the computational costs involved in the proposed scheme can be handled by sensors in WSNs.

5 Conclusions

Combining the concept of location with the merits of key-insulated systems, we introduced the first location-based data encryption for WSNs using dynamic keys. Our scheme allows a sensor node to generate a flexible ciphertext for some packet composed of many datagrams such that only the designated data aggregator has the ability to decrypt. To demonstrate the authenticity of some packet, the data aggregator is capable of revealing an ordinary signature for public verification. Our proposed protocol is conversion-free and provides unlimited time periods and random-access key-updates. In the proposed scheme, each sensor node can periodically update its private key while the corresponding public one remains unchanged. The underlining security assumption of our scheme is based on the well-known BDHP along with CDHP over elliptic curves. We also addressed detailed security proofs and precise advantage analyses to show the feasibility of our work.

Acknowledgments This work was supported in part by the Ministry of Science and Technology of Republic of China under the contract number MOST 103-2221-E-019-001.

References

1. Barreto, P., Kim, H., Bynn, B., & Scott, M. (2002). Efficient algorithms for pairing-based cryptosystems. *Advances in cryptology—CRYPTO'02* (pp. 354–368), Springer.
2. Boneh, D., & Franklin, M. (2001). Identity-based encryption from the Weil pairing. *Advances in cryptology—CRYPTO'01* (pp. 213–229), Springer.
3. Cheng, C. T., Leung, H., & Manupin, P. (2013). A delay-aware network structure for wireless sensor networks with in-network data fusion. *IEEE Sensors Journal, 13*(5), 1622–1631.
4. Cheng, C. T., Tse, C. K., & Lau, F. C. M. (2011). A delay-aware data collection network structure for wireless sensor networks. *IEEE Sensors Journal, 11*(3), 699–710.
5. Delfs, H., & Knebl, H. (2002). *Introduction to cryptography: Principles and applications.* New York: Springer.
6. Dodis, Y., Katz, J., Xu, S., & Yung, M. (2002). Key-insulated public key cryptosystems. In: L. R. Knudsen (Ed.), *Advances in cryptology—EUROCRYPT'02* (pp. 65–82). New York: Springer.
7. Dodis, Y., Katz, J., Xu, S., & Yung, M. (2003). Strong key-insulated signature schemes. In *Proceedings of public key cryptography 2003 (PKC'03)*, LNCS 2567 (pp. 167–144), Springer.
8. ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms". *IEEE Transactions on Information Theory, IT-31*(4), 469–472.
9. Hanaoka, G., Hanaoka, Y., & Imai, H. (2006) .Parallel key-insulated public key encryption. In *Proceedings of public key cryptography 2006 (PKC'06)*, LNCS 3958, pp. 105–122.
10. Hanaoka, Y., Hanaoka, G., Shikata, J., & Imai, H. (2005). Identity-based hierarchical strongly key-insulated encryption and its application. *Advances in cryptology—ASIACRYPT'05* (pp. 495–514), Springer.
11. Hsu, C. L., & Lin, H. Y. (2011). New identity-based key-insulated convertible multi-authenticated encryption scheme. *Journal of Network and Computer Applications, 34*(5), 1724–1731.
12. Huang, P. K., Lin, X., & Wang, C. C. (2013). A low-complexity congestion control and scheduling algorithm for multihop wireless networks with order-optimal per-flow delay. *IEEE/ACM Transactions on Networking, 21*(2), 495–508.
13. Intel PXA255 processor electrical, mechanical, and thermal specification. http://int.xscale-freak.com/XSDoc/PXA255/27878002.pdf
14. Ko, S. W., Yu, S. M., & Kim, S. L. (2013). The capacity of energy-constrained mobile networks with wireless power transfer. *IEEE Communications Letters, 17*(3), 529–532.
15. Lazos, L., Poovendran, R., Meadows, C., Syverson, P., & Chang, L. (2005). Preventing wormhole attacks on wireless ad hoc networks: A graph theoretic approach. In *Proceedings of IEEE wireless communications and networking conference (WCNC'05)* (pp. 1193–1199), New Orleans, LA.
16. Liao, Y., Qi, H., & Li, W. (2013). Load-balanced clustering algorithm with distributed self-organization for wireless sensor networks. *IEEE Sensors Journal, 13*(5), 1498–1506.
17. Lin, H. Y., & Hsu, C. L. (2011). A novel identity-based key-insulated convertible authenticated encryption scheme. *International Journal of Foundations of Computer Science, 22*(3), 739–756.
18. Menezes, A., Oorschot, P., & Vanstone, S. (1997). *Handbook of applied cryptography.* Boca Raton, FL: CRC Press, Inc.
19. Pointcheval, D., & Stern, J. (2000). Security arguments for digital signatures and blind signatures. *Journal of Cryptology, 13*(3), 361–396.
20. Rivest, R., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM, 21*(2), 120–126.
21. Schnorr, C. P. (1991). Efficient signature generation by smart cards. *Journal of Cryptology, 4*(3), 161–174.
22. Shamir, A. (1984). Identity-based cryptosystems and signature schemes. *Advances in cryptology—CRYPTO'84* (pp. 47–53), Springer.
23. Stallings, W. (2005). *Cryptography and network security: Principles and practices*, 4th edn., New York: Pearson.
24. Weng, J., Liu, S., Chen, K., Zheng, D., & Qiu, W. (2008). Identity-based threshold key-insulated encryption without random oracles. In *Proceedings of CT-RSA 2008*, LNCS 4964 (pp. 203–220). Heidelberg: Springer.

25. Xie, L., Zhu, H., Xu Y., & Zhu, Y.(2006). A tamper-resistance key pre-distribution scheme for wireless sensor networks. In *Proceedings of 5th international conference on grid and cooperative computing workshops (GCCW'06)* (pp. 437–443).
26. Yu, C. W., Tseng Y. M., & Wu, T. Y. (2010). A new key-insulated signature and its novel application. In *Proceedings of cryptology and information security conference (CISC 2010)*, Taiwan.
27. Zhang, Y. C., Liu, W., Lou, W. J., & Fang, Y. G. (2006). Location-based compromise-tolerant security mechanisms for wireless sensor networks. *IEEE Journal on Selected Areas in Communications, 24*(2), 247–260.
28. Zhou, Y., Cao, Z., & Chai, Z. (2006). Identity based key insulated signature. In *Proceedings of ISPEC 2006*, LNCS 3903 (pp. 226–234).

Han-Yu Lin received B.A. degree in economics from the Fu-Jen University, Taiwan in June 2001, his M.S. degree in information management from the Huafan University, Taiwan in June 2003, and his Ph.D. degree in computer science and engineering from the National Chiao Tung University, Taiwan in December 2010. He served as a research assistant in the Department of Computer Science and Engineering, National Taiwan Ocean University, Taiwan from March 2011 to December 2011. He was a senior engineer in CyberTrust Technology Institute, Institute for Information Industry, Taiwan from January 2012 to July 2012. Since August 2012, he has been an Assistant Professor in the Department of Computer Science and Engineering, National Taiwan Ocean University, Taiwan. His research interests include cryptology, network security, digital forensics, RFID privacy and application, cloud computing security and e-commerce security.