doi:10.3772/j.issn.1006-6748.2017.01.006

Research on stateful public key based secure data aggregation model for wireless sensor networks①

Qin Danyang (秦丹阳)②, Jia Shuang, Yang Songxiang, Wang Erfu, Ding Qun
(Key Lab of Electronic and Communication Engineering, Heilongjiang University, Harbin 150080, P. R. China)

Abstract

Data aggregation technology reduces traffic overhead of wireless sensor network and extends effective working time of the network, yet continued operation of wireless sensor networks increases the probability of aggregation nodes being captured and probability of aggregated data being tampered. Thus it will seriously affect the security performance of the network. For network security issues, a stateful public key based SDAM (secure data aggregation model) is proposed for wireless sensor networks (WSNs), which employs a new stateful public key encryption to provide efficient end-to-end security. Moreover, the security aggregation model will not impose any bound on the aggregation function property, so as to realize the low cost and high security level at the same time.

Key words: wireless sensor networks (WSNs), secure data aggregation, homomorphic encryption, simple power analysis

0 Introduction

WSN (wireless sensor network) is a major branch of the Internet of things, which is made up of many sensor nodes with constrained resource. Sensors will be affected by power supply, communication and computing power, so the data transmission is restricted[1]. Data aggregation is thought to be essential technique for WSN, since it may save the computation and communication energy effectively. With such a technique, data will be captured by sensor nodes and fused by intermediate nodes and then transmitted to sink nodes through the wireless link. Under the background of the Internet of things, the aggregation node is regarded as a network drive to collect aggregated data and send them to the cloud[2]. Data aggregation adopted in WSN will reduce packet transmission and data redundancy, and will improve the overall lifetime.

Traditional security aggregation protocol adopts single hop encryption, in which sensor nodes will encrypt the captured data and send the ciphertext to the aggregation node; then the aggregation node will decrypt the received data, perform the aggregation function, and finally send the encryption result to the upper aggregation node. Therefore, the data aggregation protocol improves the bandwidth and network energy efficiency, but at the same time brings higher computational overhead and latency. What is more, the aggregation node may access to the plaintext data, which means there is no guarantee for end-to-end confidentiality[3]. Therefore, a higher security and more efficient solution is needed for WSN.

1 Research status on data aggregation

In recent years, a number of security schemes to guarantee the data confidentiality without causing delay are proposed, such as CDA (concealed data aggregation) based on PH (privacy homomorphism) which enables direct calculations on enciphered data. The advantage of CDA is confidentiality of the data without complex computation. PH, however, will significantly increase the demand for encryption energy, and limit the applicability of the aggregation query as well[4]. ECEG (Elliptic Curve El Gamal) is one of the most popular algorithms for WSN currently, which can be implemented on MicaZ efficiently with unit execution time as 1.29s. However, in some condition, the sink node needs to collect information about the target area continuously about every 20s, so such a scheme is impracticable and will cause large energy consumption.

① Support by the National High Technology Research and Development Program of China (No. 2012AA120802), the National Natural Science Foundation of China (No. 61302074), Specialized Research Fund for the Doctoral Program of Higher Education (No. 20122301120004), Natural Science Foundation of Heilongjiang Province (No. QC2013C061).

② To whom correspondence should be addressed. E-mail: qindanyang@hlju.edu.cn

Received on Mar. 8, 2016

In addition, ECEG is an additive homomorphic and hence, it can only support a limited number of aggregation functions related to the addition operation. Moreover, since a single hop authentication could not guarantee the aggregator to perform the aggregate function properly and a compromised aggregator may produce fake aggregation to authenticate with its legitimate key, the end-to-end data integrity as a new security requirement becomes a new challenge for WSN nowadays, which will be considered as well in the research.

2 Key encryption and attacks

The SDAM adopts the combination of asymmetric and symmetric encryption algorithm, so it will briefly introduce the encryption algorithm and the classical attack types in this section.

2.1 Stateful public key encryption

The stateful encryption can significantly reduce the computational cost of traditional PKE (public key encryption). In SPKE (stateful public key encryption), the sender uses a state repeatedly in different encryption algorithms, which is held by the sender, and the same state is reused in different encryption processes. The use of stateful encryption will reduce the computational cost and the energy for being calculated only once. In this work, SPKE is utilized to guarantee the convergecast traffic toward BS efficiently. Accordingly the state is adopted to share with BS the information that the network nodes use to ensure the end-to-end security for aggregated data.

2.2 Cryptographic methods

2.2.1 Homomorphic encryption

HE (homomorphic encryption) allows direct calculation on ciphertext with the same effect as that on the underlying plaintext data. An encryption algorithm is considered to be homomorphic, if and only if there is

$$E(m_1 \odot_M m_2) \leftarrow E(m_1) \odot_C E(m_2), \quad \forall m_1, m_2 \in M \qquad (1)$$

where M and C are the set of plaintext and the set of ciphertext respectively; the operation $\odot$ may support the addition and multiplication or support these two kinds of algorithms at the same time with the choice depending on the characteristics of the encryption scheme. The one supporting all the functional operation on the ciphertext is known as FHE (fully homomorphic encryption). The other class of HE is PHE (partially homomorphic encryption), which includes the encryption schemes that have homomorphic property.

2.2.2 Message authentication code

It is clear that MAC (message authentication code) doesn't satisfy the additive property:

$$\mathrm{MAC}(\alpha + \beta) \neq \mathrm{MAC}(\alpha) + \mathrm{MAC}(\beta) \qquad (2)$$

where MAC can be aggregated by XOR (exclusive operation) as Eq. (3) to satisfy the integrity and authenticity so as to verify all the personal data.

$$\mathrm{MAC}_{agg} = \mathrm{MAC}_1 \oplus \mathrm{MAC}_2 \oplus \cdots \oplus \mathrm{MAC}_n \qquad (3)$$

where HMAC (Hash-based MAC) is a kind of scheme based on encryption hash function, which is always used to validate data integrity and source. The encryption intensity of HMAC depends on the property of the underlying hash function. Let HMAC(K, m) denote the message input of m using a key with the assumption that the underlying hash function is SHA-1, which will produce 20-bytes digest as the output.

2.2.3 Relative functions

There will be two main relative functions to be adopted in the research. PRF (pseudo random function) is a deterministic function with two inputs, namely K and m, where K is a hidden key and m is a random variable. Generally, the output of PRF by computing is different from the real output value. KDF (key derivation function) takes a given key and a random number as the input, and will generate a new key for the encryption algorithm. In this study, the HKDF (key derivation function based on Hash message authentication code) NISTSP800-108 is adopted to generate a dynamic key.

2.3 Network model

WSN is composed of a large number of sensor nodes and BS. Sensor nodes are resource-constrained devices, deployed in a geographical area to sense and monitor. They are arranged in multiple clusters, as shown in Fig. 1. Some sensor nodes are chosen as the CHs (cluster-heads) based on dynamic selection algorithm to aggregate data from their members, and will forward the results to the next hop. All the notations used in this paper are shown in Table 1.

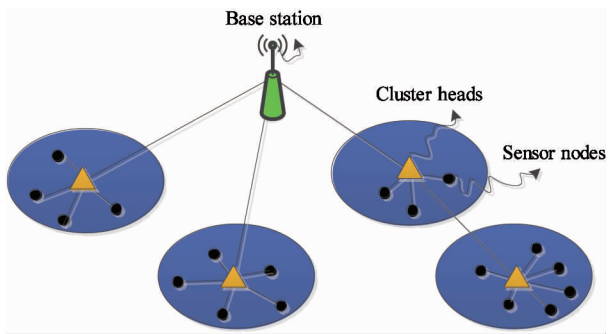

Fig. 1 Wireless sensor network model

Table 1 Notations and definitions

Notation	Definition	Notation	Definition
NR	Number of sensor nodes	Y	The base station's public key
R	Number of cluster heads in the network	$\|0^z$	Concatenation with a serial of z '0's
CH_j	Cluster head CH_j, $j \in \{1,\cdots,R\}$	λ	Number of bits needed to represent the data captured
L	Maximum number of nodes per cluster	SK_{ij}^{BS}	The key shared between S_{ij} and BS
BS	The base station	N_{ij}	A sequence number for data freshness
S_{ij}	Sensor node belonging to CH_j, $j = \{1,\cdots,L\}$	SN	Sensor nodes

2.4 Attack model

The attack types in WSN can be categorized as follows: (i) An attacker can eavesdrop and monitor the transmitted data in wireless sensor network; (ii) An attacker can produce an illegal data to modify the transmitted packets, and replay the sent packets; (iii) An attacker can threaten the sensor nodes covertly using power analysis.

The first category mainly focuses on the passive attack to derive the key, as the main purpose. The basic attack is CA (cipher analysis), with such attack that an attacker can only obtain the information by explaining the ciphertext. Then the attacker will start KPA (known plaintext attack). In KPA, an attacker will attempt to infer the secret information with a known plaintext and corresponding ciphertext. In the same category, an attacker could also choose any plaintext to encrypt and study the generated ciphertext, namely CPA (chosen plaintext attack). In the context of WSN, the practicability of CPA is considered to be less practical than CA and KPA, but it is still a very dangerous attack. In the second category, an attacker will interfere with communication frequently and actively. Active attack includes extensibility, forge packets and playback. Extensibility allows an attacker to modify the packet without any knowledge of the content. The homomorphic encryption possesses the extensibility itself[5]. An attacker may forge the packets, and even replay valid packets already "used" in the WSN in order to deceive BS. In the third category, an attacker can covertly perform channel attacks. Such attack allows an attacker to use information leakage in the process of cryptographic operations to obtain all or a part of the secret keys[6]. In this category, an attacker does not need to delete or interfere with the operation of a node from the network, but can take power traces. In other words, an attacker can perform SPA (simple power analysis), which is quite common in WSN.

3 Secure data aggregation models

SDAM proposed in this work consists of two main phases, namely the forwarding phase and the aggregation phase. In the former phase, all the sensors will send their states for being used in the aggregation phase. In the latter phase, the sensor nodes will encrypt and verify the captured data using the states shared with BS. Then, CH will use homomorphic operation and XOR to aggregate ciphertext and signature respectively in order to generate a cipher and a new signature. Finally, BS will verify the aggregated data, decrypt aggregation, retrieve plaintext and call the verification process.

3.1 Parameters setting

ECC (error correction code) for SPKE is an effective secure method. Suppose BS generates a pair of keys (x, Y) before deployment, where there is the relationship between the public key Y and the private key x as $Y = xG$. Each sensor S_{ij} carries the keys SK_{ij}^{BS} shared with BS. Elliptic curve parameters are set as (Y, E, p, G, n), where Y is the elliptic curve public key; E is the elliptic curve over Galois field Z_p with p as a sufficiently large enough prime number; G is the base point of E; and n is the order of G. Sensors are also loaded with integer M, PRF based KDF (NIST SP800-108 HKDF) and safe MAC (HMAC).

3.2 Forwarding phase

In the forwarding phase, sensor nodes will send the state St_{ij} to produce keys required by the aggregation phase. HKDF is adopted to obtain the authentication keys. In fact, the output K_{ij} of the PRF is computed with a nonce as the iteration variable, and then is used as key material for authentication. Each sensor S_{ij} executes Algorithm 1 to send output St_{ij} and MAC_{ij} to the next hop. In order to extract all the keys from base stations, all packets must be sent. Therefore, at such stage, CH acts as a data forwarder but not a data aggregator, as shown in Fig. 2(a).

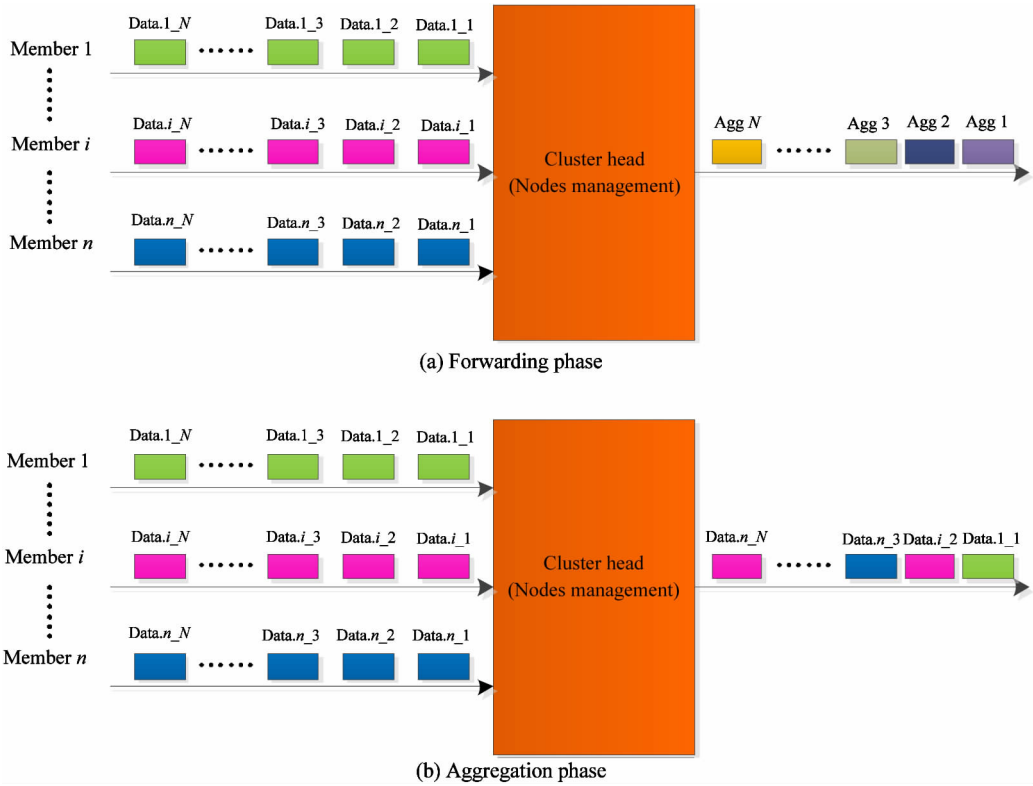

Fig. 2 Data transmission in SDAM

Algorithm 1: Forwarding phase (S_{ij})

Input: (Y, E, p, G, n), SK_{ij}^{BS}, Nonce

1. Generate a random $r_{ij} \in [1, n-1]$
2. Compute $St_{ij} = r_{ij}G$
3. Compute $K_{ij} = HKDF(St_{ij} \| r_{ij}Y \| SK_{ij}^{BS}, N_{ij})$
4. Compute $MAC_{ij} = HMAC(St_{ij}, K_{ij})$

Output: St_{ij}, MAC_{ij}

Each sensor will keep the state (r_{ij}, St_{ij}) and use the state to encrypt. Once the state is received, CH forwards all data to BS or to the nearest CH. Then, BS will verify the integrity by using private key x to identify the entire senders. The verification is done by calculating all the keys corresponding to the received states (see Algorithm 2). If the verification is established, the corresponding state St_{ij} will be stored in the BS's database, which is used for decryption and verification. Otherwise, the state will be rejected.

Algorithm 2: Verification (BS)

Input: (Y, E, p, G, n), SK_{ij}^{BS}, x, Nonce, all pairs (St_{ij}, MAC_{ij})

1. **For** each $i \in \{1, \cdots, L\}$, $j \in \{1, \cdots, R\}$
1.1. Compute $K_{ij} = HKDF(St_{ij} \| x_{ij}St_{ij} \| SK_{ij}^{BS}, N_{ij})$
2. **For** each $i \in \{1, \cdots, L\}$, $j \in \{1, \cdots, R\}$
2.1. Compute $MAC_{ij}' = HMAC(St_{ij}, K_{ij})$
2.2. **If** $MAC_{ij}' = MAC_{ij}$,
Then accept
Otherwise reject

Output: MAC verification

3.3 Aggregation phase

The aggregation phase consists of three steps: encryption, aggregation and verification.

3.3.1 Encryption

In this step, data m_{ij} is captured by S_{ij}, and it is encoded before encryption, with the resulting code being encrypted by symmetric encryption. In the aggregation phase, HKDF will generate two keys, namely K_{ij1} and K_{ij2}. Sensors encrypt the encoded plaintext and use K_{ij1} and K_{ij2} to calculate the corresponding MAC respectively. The encryption is performed using addition modulo the large number M (see Algorithm 3), where M must be greater than $e_{agg} = \sum_{i=1\cdots L}^{j} e + ij$. If this property is verified, the decryption will result in a message e_{agg} that is smaller than M. The nonce N_{ij} used in HKDF ensures the dynamic keys needed for the security of encryptions. The MAC is then calculated on ciphertext. Finally, the ciphertext and MAC are sent to the corresponding CH.

Algorithm 3: Encrypt&sign (S_{ij})

Input: m_{ij}, (r_{ij}, St_{ij}), Y, M, Nonce

1. Encode m_{ij} into $e_{ij} = m_{ij} \| 0^z$, where $z = \lambda \cdot (i-1)$
2. Compute $K_{ij} = HKDF(St_{ij} \| r_{ij}Y \| Y, N_{ij})$, where $K_{ij} = K_{ij1} \| K_{ij2}$
3. Compute $C_{ij} = K_{ij1} + e_{ij} \bmod M$
4. Compute $MAC_{ij} = HMAC(C_{ij}, K_{ij2})$

Output: C_{ij}, MAC_{ij}

3.3.2 Aggregation

In this step, CH acts as the data aggregator and uses dispersion aggregation. For multi-user, the data are randomly aggregated, the process of which is shown in Fig. 2(b). CH will aggregate all ciphertext including its own ciphertext into a new cipher C_{agg}, and aggregate MACs and its own MAC into a new MAC_{agg} (see Algorithm 4). Using addition operation modulo, the ciphertext is aggregated homomorphically, and MAC is XORed[7]. After that, the output of Algorithm 4 will be sent to BS or the nearest CH. When a CH receives the packets from another CH, it will forward the packet to BS. The execution homomorphism aggregation of each CH_j is as follows:

Algorithm 4: Homomorphic aggregation (CH_j)

Input: All pairs (C_{ij}, MAC_{ij}), where $i \in \{1, \cdots, L\}$

1. **For** L ciphertexts ($C_{1j} \cdots C_{Lj}$)

1.1. Compute $C_{agg} = \sum_{i=1\cdots L}^{j} C_{ij} \bmod M$

2. **For** LMACs($\mathrm{MAC}_{1j} \cdots \mathrm{MAC}_{Lj}$)

2.1. Compute $\mathrm{MAC}_{agg} = \oplus \mathrm{MAC}_{ij}$

Output: C_{agg}, MAC_{agg}

3.3.3 Verification

In the verification step, BS will call the decryption and verification process for the aggregation of each cluster after the data packets are received. BS will firstly calculate the current keys corresponding to all network nodes using the state stored in the database. Then, BS will decrypt the aggregated ciphertext, and retrieve the personal plaintext (see Algorithm 5). Finally, BS will calculate (C_{ij}, MAC_{ij}), and check the end-to-end integrity of the information. If the verification is passed, aggregated data e_{agg} will be accepted; otherwise it will be rejected. BS will send (C_{ij}, MAC_{ij}) by notifying CH_j, and verify each pair of nodes to determine the malicious nodes. Another advantage of the model is that the node does not need to send a response to BS, because all the sensors are involved in the aggregation. In SDAM, even without sensing data, each sensor will produce the encryption and sign MAC. In Algorithm 5, the non-responding node will simply use the zero value of m; thus, after sending all the sensor data to BS, it can perform any aggregation function, which is a major advantage of multi-hop solutions. SDAM is flexible without imposing any constraints on the property of the function.

Algorithm 5: End-to-end (BS)

Input: All pairs (C_{agg}, MAC_{agg}), where $j \in \{1, \cdots, R\}$

1. Compute $K_{ij} = HKDF(St_{ij} \| x_{ij}St_{ij} \| Y, N_{ij})$
2. **For** each pair (C_{agg}, MAC_{agg})$_j$

2.1. Compute $e_{agg} = C_{agg} - \sum_{i=1\cdots L}^{j} K_{i1} \bmod M$

2.2. Encode (e_{agg}, L, λ): $m_i = e[(i-1) \cdot \lambda, \lambda \cdot i - 1]$, where $i = 1, \cdots, L$

2.3. **For** m_i, where $i = 1, \cdots, L$

2.3.1. Compute MAC_i

2.4. Compute $\mathrm{MAC}'_{agg} = \oplus \mathrm{MAC}_i$

2.5. **If** $\mathrm{MAC}'_{agg} = \mathrm{MAC}_{agg}$

Then e_{agg} is accepted

Otherwise e_{agg} is rejected

Output: MAC verification

Aggregation phase is to be executed for many times until the state is failed. Therefore, a deadline has been designed in this paper. Key expiration is a very important security measurement, which allows key refreshing and involves a new forwarding phase so as to increase the security theoretically. In SDAM, a node can be seen as a lifetime sequence epoch, and each period consists of two phases, namely forwarding and aggregation phases (see Fig. 3). In fact, the security level can be used to judge whether to meet the required security level of sensitive information. Therefore, the key refreshing is very important for security improvement before deadline, which may be pre-installed in the sensor practically. Further, due to some faults, some nodes' synchronization may be lost. Then, the synchronization of new forwarding phase should be refreshed. In this case, the base station will request a new forwarding phase by using a specific active message.

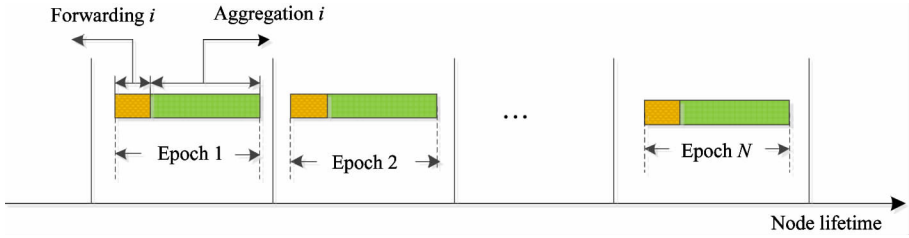

Fig. 3 Node lifetime

4 Simulation results and performance analysis

The confidentiality and integrity, as well as the performance evaluation of SDAM will be analyzed in terms of security, computation and communication overhead, energy consumption, scalability and portability.

4.1 Security performance analysis

SDAM proposed in this paper will increase the data aggregating efficiency and enhance the information security performance at the same time. The metric of average packet successful delivery rate (PSDR)[8] with different number of attackers is adopted to compare different security algorithms. The number of nodes remains 200 with the attack type as the classical SPA (simple power analysis), and the security aggregating algorithms are SHA[9], EVCDA[10] and SDAM. The simulating results in Fig. 4 show that the average PSDR will decrease with the number of attackers increasing for all the algorithms. The average PSDR of SDAM, however, is superior to the other two obviously. Especially when the number of attackers approaches to 10% of the total nodes, SHA and EVCDA are not available, while it is still over 40% with SDAM. The error packets are reduced by SDAM since the bidirectional malware detection technology to eliminate malicious node cluster members and CH. Therefore, the security performance can be well provided by SDAM.

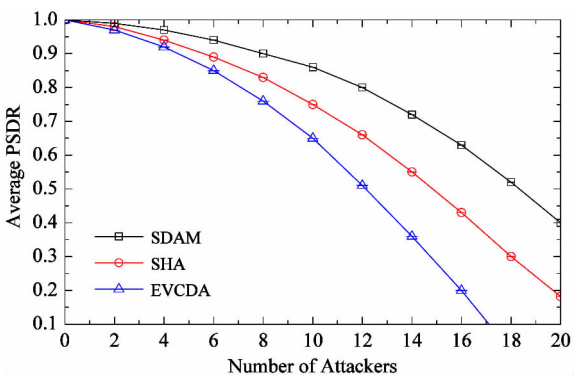

Fig. 4 Comparison of average PSDR with different number of attackers

4.2 Computation overhead analysis

To analyze the computational complexity reduction, let SM denote the cost of scalar multiplication, MA denote the cost of modulo addition, and SG denote the cost of one signature. In the forwarding phase, each sensor must calculate its state and send it to BS. Such calculation will involve $2SM + 2SG$ operations. In the aggregation stage, each sensor requires an operator to calculate the encryption and signature for $MA + 2SG$ operations. It will cost $CH_2 \cdot (L-1) \cdot MA$ to perform homomorphic aggregation. Thus, the total computational cost of SDAM will be:

$$2SM + 2SG + NS \cdot (NR \cdot (MA + 2SG) + 2R \cdot MA \cdot (L-1))$$

where NS is the number of session in one aggregation phase.

In order to illustrate it obviously, the operations are implemented on TelosB, with TinyOS and TinyECC being adopted. TinyOS is an open resource designed for low-power wireless equipment, while TinyECC is a free library that provides operations of elliptic curves over prime field F_p[11]. For HKDF and signature, HMAC provided by the library is used and SHA-1 is taken as hash function. HKDF will generate 20-byte key in forwarding phase and two 20-byte keys in aggregation phase. Here M (see Section 3.1) should be selected as 2^{160} to avoid overflow. Before calculating the cost of encryption function, it is focused on the scalar point multiplication, efficiency and security against side channel attacks. Table 2 shows the execution time of the proposed encryption functions on TelosB, where

Table 2 Execution time of the proposed cryptographic functions

Cryptographic		Execution time(s)	SPA
Stpke. state( )	SWM($w=4$)	5.82	Yes
	Comb($w=4$)	2.71	Yes
	Sec. Comb($w=4$)	2.85	No
	Sec. Comb($w=5$)	2.29	No
Stpke. encrypt( )		0.081	No
Hom. add( )		0.002	No

both SWM (sliding windows method) and Comb method are cryptographic functions to improve the performance of *SM*.

w in Table 2 is the number of windows being used. G and Y are pre-calculated in *Stpke. init*( ). The execution time is the average execution time of the relative operations. Since the curve points (G and Y) are fixed, the overhead of *Stpke. init*( ) can be ignored. SPA attacks only consider the state calculation, namely *Stpke. state*( ). And the state of SDAM is used in the aggregation phase and performs unprotected operation; SPA attacker can restore the state, thus undermining the aggregation of all communication phases. However, the keys used for encryption and authentication will change from one message to another. Thus, the operations performed in aggregation phase namely, *Stpke. encrypt*( ) and *Hom. Add*( ) are not vulnerable to SPA. The results in Table 2 show that, in the forwarding phase, the comb method can significantly increase the perform time. In addition, SPA secure version is efficient, and it will improve the operating speed by 61% and 15% faster than SWM and Comb, both of which are vulnerable to SPA attacks.

In the aggregation phase, the sensor takes about 0.081s to encrypt and generate an MAC. Comparison with related encryption methods shows that SDAM is much more effective since the time is cost only in the forwarding phase. In the aggregation phase, the aggregation state performs homomorphism operations on encrypted data with only a small amount of computation. In other models, the aggregation should be operated on the elliptic curve, which will not only produce high energy consumption, but also increase the end-to-end delay[12], for BS must wait for a certain period of time before receiving the aggregate data. SDAM takes the advantages of ECC and El Gamal encryptions to reduce the computational overhead. Moreover, El Gamal scheme is adopted to produce a state, which will be taken into encryption in all future transmissions to further reduce the operating overhead.

4.3 Communication overhead analysis

In the forwarding phase, the communication complexity is $O(1)$ for the non-CH nodes. For CH node, however, the complexity is $O(L)$. Since all the data packets are sent to BS, the total communication overhead at this stage is $R(2L-1)$. In the aggregation phase, the data are aggregating toward the BS to which each sensor sends a data packet to form aggregating communication streams. The complexity of both non-CH and CH node are $O(1)$. Thus, the total communication cost is N. It only considers the condition that CH is directly connected to BS. Otherwise the costs may increase depending on the depth and the number L.

Simulations take three models as SHA, EVCDA and SDAM to compare. In simulation scenarios, 50, 100 and 150 sensor nodes are randomly deployed within a square area with a single base station centered. In SDAM, the state is a point belonging to an elliptic curve, which is transmitted in a compressed way and requires only $(1+\log_2 p)$ bits. In the aggregation phase, a 20-byte encoded plaintext is used to produce a 20-byte ciphertext. Therefore, short packets are only adopted to provide both end-to-end confidentiality and integrity. The cost can be calculated by the following ways for transmitting a corresponding data packet, which corresponds to two different methods: (i) sending longer messages, which will increase the bit error rate and reduce the reliability; (ii) splitting the packet into blocks and sending each block separately which will incur not only delay but also additional cluster head overhead. However, no matter which is used, SDAM will produce less energy consumption. Simulating duration is 500s for each. The average of 10 simulations will be obtained. The data are sent to BS using a simple clustering algorithm based on TDMA. 4, 8 and 12 cluster heads (channels) are selected for three topologies with number of nodes. Therefore, the number of nodes per cluster is $L \in [8,13]$. It is noted that for the same simulation time, an expiration date is considered 5 times in 5E (5 Epochs) and 10 times in 10E (10 Epochs). The results of different models are shown in Fig. 5. Simulation results show that SDAM will bring lower communication overhead. Due to the use of the stateful public key encryption, the data using symmetric encryption can produce short ciphertext, resulting in short packets. In addition, the advantage of SDAM can be viewed when the network density is increasing as shown in Fig. 5 (b) and (c).

4.4 Energy consumption analysis

Energy is the core issue of WSN[13]. Computation and communication have a direct impact on the energy consumption and the lifetime of nodes[14]. TOSSIM-CC2420 integrates a Power-TOSSIM. The dynamic model is extended to TOSSIM. Power-TOSSIM includes the model of TelosB power consumption. However, TOSSIM cannot simulate the execution time of CPU, and will not provide any accurate information for the energy consumption calculation. For this purpose, the computation overhead corresponding to each transmitted/received packet is added in performance analysis. Energy consumption E can be calculated by $E = U \cdot I \cdot t$,

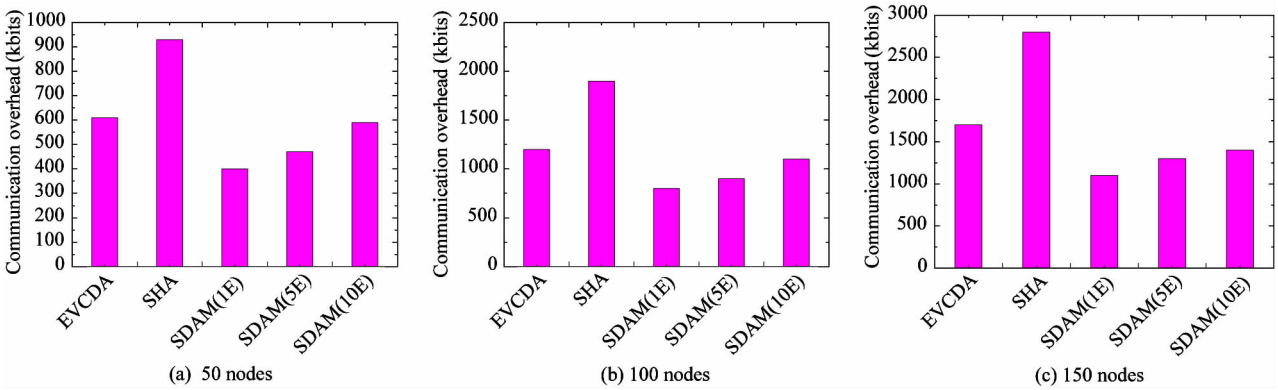

Fig. 5 The communication overhead for different topologies

where U denotes the voltage on the calculation, I denotes current, and t denotes the execution time. For 2AA batteries, the voltage is 3.0V, and the current consumption of TelosB is 1.8mA for MCU On/Radio Off. Thus the energy consumption of the proposed cryptographic function can be calculated and is given in Table 3.

Table 3 Energy consumption of the proposed cryptographic functions

Cryptographic function	Energy consumption (mJ)
Stpke. state( )	12.366
Stpke. encrypt( )	0.437
Hom. add( )	0.009

These measurements are added to the energy model. The simulations with three models above are performed with 500s for each to estimate the energy consumption in the entire network. The results at non-CH and CH node are shown in Fig. 6(a) and (b) respectively.

In Fig. 6(a), the comparison shows that SDAM will greatly reduce the energy consumption, due to the symmetric primitive and the asymmetric operations (state calculation) in SDAM which will cause less computational overhead. Thus, for providing the same level of security, SDAM will increase the network lifetime obviouly. Fig. 6(b) shows that, no matter at CH or at non-CH, there will be significant reduction in SDAM. That is because (i) the nodes will continuously perform a large amount of calculation in EVCDA, while the complex operations of SHA lie in forwarding phase; (ii) CH will participate in aggregation and will perform an aggregate function (same-state operation). In SHA and EVCDA, the addition operation is required on the elliptic curve, while it will only demand a small amount of calculation in SDAM.

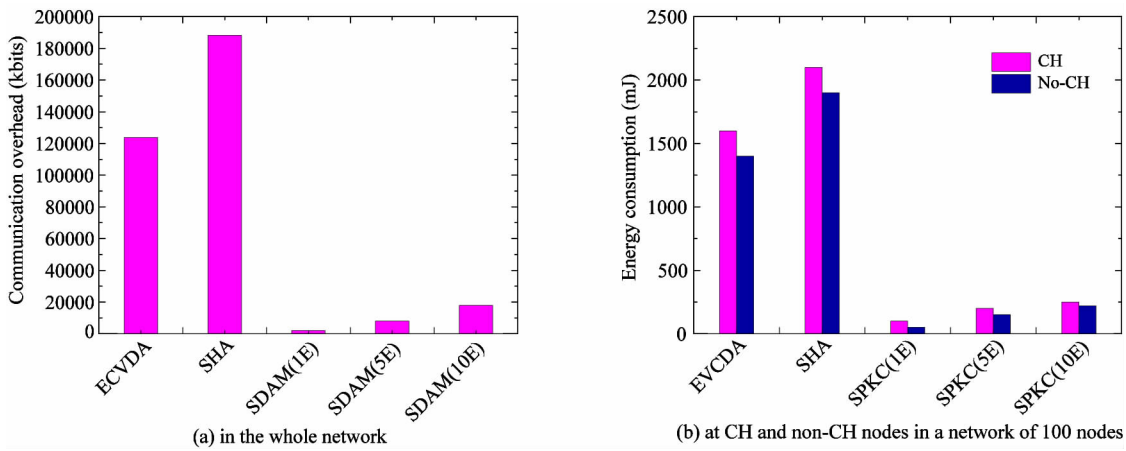

Fig. 6 The estimated total energy

It follows that the energy cost of communication using asymmetric calculation can be ignored. However, if the energy required for cryptographic computations is reduced, the communication cost will become crucial. By saving two ECC scalar multiplications, SDAM will reduce the computational cost of the traditional PKE. Therefore, the communication cost is considered to highlight the advantages of using data aggre-

gation in WSN. The estimated total energy costs of SHA, EVCDA and SDAM at the CH node and non-CH node are presented in Table 4. The results show that the computation cost on TelosB is reduced by SDAM effectively so as to lower the total energy cost.

Table 4 The estimated total energy costs (mJ) of SHA, EVCDA and SDAM at CH and non-CH node in a network of 100 nodes

Scheme		CH	non-CH
SDAM (1E)	Comm	33.446(60%)	15.029(48%)
	Comp	21.999(40%)	16.649(48%)
	Total	55.465	31.678
SDAM (5E)	Comm	35.711(32%)	15.191(20%)
	Comp	77.461(68%)	63.038(80%)
	Total	113.172	78.229
SDAM (10E)	Comm	38.103(21%)	15.284(11%)
	Comp	141.145(79%)	128.369(89%)
	Total	179.248	143.653
EVCDA	Comm	39.007(3%)	16.748(1%)
	Comp	1497.143(97%)	1408.692(99%)
	Total	1536.15	1425.44
SHA	Comm	90.744(4%)	27.488(1%)
	Comp	2038.263(96%)	1950.672(99%)
	Total	2129.007	1978.16

4.5 Scaling and portability analysis

The solution of SDAM is scalable and can be added to any desired cluster. The only condition is that the number of nodes L per cluster cannot exceed the maximum number of nodes supported by the coding function. How to change the level of security and the number λ and L is shown in Table 5. For multi-hop network, CH that received the aggregation ciphertext from another CH needs to send the corresponding data packets to the BS or the nearest CH.

Table 5 Maximum number of nodes per cluster and the security level

Security level	λ	L
80 bits	1	80
	2	40
	4	20
	8	10
160 bits	1	160
	2	80
	4	40
	8	20

SDAM has also been implemented on MicaZ mote platform . In fact , the state calculation requires only 1.48s by ECC, encrypt and sign function takes approximate 0.057s, and the homomorphism operation requires only 0.0012s. The corresponding energy consumption is calculated and shown in Table 6. Since power consumption of TelosB is lower than MicaZ, its total energy consumption is low as well. In fact, the results show that the execution time of MicaZ is 34% faster than that of TelsoB, while in the same period the energy consumption of TelsoB is only 27% of that of MicaZ.

Table 6 Estimated energy costs of SDAM computation on MicaZ and TelsoB motes

Cryptographic function		MicaZ	TelosB
Stpke. state( )	Time	1.48s	2.29s
	Energy	44.24mJ	12.366mJ
Stpke. encrypt( )	Time	0.057s	0.081s
	Energy	1.7mJ	0.437mJ
Hom. add( )	Time	0.0012s	0.002s
	Energy	0.036mJ	0.009mJ

5 Conclusion

Wireless sensor network is an important component of modern communication systems, and the security of the network is an important guarantee for the successful rate of data transmission, so the study of WSN security is very important and necessary. In this study, a secure data aggregation model (SDAM) based on stateful public key is proposed, which uses an addition homomorphic encryption and aggregated MAC to provide end-to-end confidentiality and integrity. Experimental and simulation results have confirmed the efficiency of SDAM in terms of security and energy, comparing with traditional models. Moreover, relative data and curves illustrate that SDAM can achieve higher security level and produce less energy consumption. In the topology of the 100 nodes, SDAM generates energy consumption in the cluster head node only 179.248mJ, while the energy consumption is 1536.15mJ for EVCDA and 2129.007mJ for SHA. Therefore, SDAM is able to achieve an efficient low-power and safe data aggregation. Future research will extend to the network with moving nodes and will consider new attacks such as selective forwarding in order to provide the best way for subsequent research on aggregated data. Some of the results will provide ideas for robust multihop routing in future ubiquitous communication network.

References

[1] Li F M, L X H, Kuang H L. Reaearch on an energy-efficient low latency flooding algorithm for wireless sensor network. *Journal of communication*, 2013, 28(8): 46-53

[2] Habib M, Ammari B. On the problem of k-coverage in mission-oriented mobile wireless sensor networks. *Computer Networks*, 2012, 6(1):7-9

[3] Chen Z Y, Yang G, Chen L, et al. Summary of wireless sensor network data fusion research. *Application Research of Computers*, 2011, 10(5): 6-8

[4] Lucas D, Joel J. A survey on cross-layer solutions for wireless sensor networks. *IEICE Transaction on Journal of Network and Computer Applications*, 2011, 5(34): 523-534

[5] Egemen K C, Dan B, Amit D, et al. Modelling communication network challenges for future internet resilience, survivability, and disruption tolerance: a simulation-based approach. *Telecommunication Systems*, 2014, 2(6): 751-768

[6] Liu M, Gong H G, Mao Y C. Collection and aggregation protocol on high efficiency and energy saving of sensor network data. *Journal of software*, 2012, 16(12): 2106-2116

[7] Kumar V, Madria S K. Secure hierarchical data aggregation in wireless sensor networks: performance evaluation and analysis. In: Proceeding of the IEEE 14th International Conference on Mobile Data Management (MDM), Bengaluru, India, 2012. 196-201

[8] Rabaey J, Ammer J. Distributed framework for correlated data gathering in sensor networks. *IEEE Transactions on Mobile Computing*, 2010, 57(1): 578-593

[9] Albath J, Madria S K. Secure hierarchical data aggregation in wireless sensor networks. In: Proceeding of the Wireless Communications and Networking Conference (WCNC), Bengaluru, India, 2009. 1-6

[10] Sun H M, Lin Y H, Hsiao Y C, et al. An efficient and verifiable concealed data aggregation scheme in wireless sensor networks. In: Proceeding of the International Conference on Embedded Software and Systems, Chengdu, China, 2008. 19-26

[11] Yao J S, Liu Y L. WSN hierarchical trust management model. *Heilongjiang science and technology information*, 2013, 36(11): 25-26

[12] Sanli H O, Ozdemir S, Cam H. SRDA: secure reference-based data aggregation protocol for wireless sensor networks. In: Proceeding of IEEE 60th Vehicular Technology Conference (VTC), Los Angeles, USA, 2004. 4650-4654

[13] Hedabou M, Beneteaus L, Pinel P. Some ways to secure elliptic curve cryptosystem. *Advances in Applied Clifford Algebras*, 2008, 8(2): 677-688

[14] Anastasi G, Conti M. Data collection in sensor networks with data mules: An intergrade simulation analysis. *IEEE Symposium on Computers and Communications*, 2012, 3(7): 1096-1102

Qin Danyang, born in 1983. She received her B. Sc. degree in communication engineering from Harbin Institute of Technology in 2006, and both M. Sc and Ph. D. degrees in information and communication system from Harbin Institute of Technology in 2008 and 2011 respectively. Currently, she is an associated professor at the Department of Communication Engineering of Heilongjiang University, Harbin, P. R. China. Her researches include wireless sensor network, wireless multihop routing and ubiquitous sensing.