This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2010 proceedings

TESP^2: Timed Efficient Source Privacy Preservation Scheme for Wireless Sensor Networks

Rongxing Lu[†], Xiaodong Lin[‡], Haojin Zhu[§], and Xuemin (Sherman) Shen[†]
[†]Dept. of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
[‡]Faculty of Business and Information Technology, University of Ontario Institute of Technology, Ontario, Canada, Oshawa, Ontario, Canada L1H 7K4
[§]Dept. of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
Email: {rxlu, xshen}@bbcr.uwaterloo.ca; xiaodong.lin@uoit.ca; zhu-hj@cs.sjtu.edu.cn

***Abstract*—Source privacy preservation against global eavesdroppers' traffic analysis attack is one of the most challenge issues in wireless sensor networks. In this paper, we present a new timed efficient source privacy preservation (TESP^2) scheme. In the TESP^2 scheme, each sensor node broadcasts timed data collection request to its upstream nodes, and then each upstream node will return the real data's ciphertext if it has detected something, or a dummy data's ciphertext if it hasn't. After receiving ciphertexts from upstream nodes, the sensor node will filter the dummy data, re-encrypt and forward the real data's ciphertexts to its downstream node to achieve the source privacy preservation. Security analysis and extensive simulation results demonstrate the proposed TESP^2 scheme can resist the traffic analysis attack and achieve high source privacy preservation with some tolerant latency.**

***Keywords*— Wireless sensor networks, security, source privacy preservation, timed data collection, re-encryption technique**

I. Introduction

Recent advances in micro electro mechanical systems and wireless communication have paved the way for the rapid deployment of wireless sensor networks, which have been well recognized as a ubiquitous and general approach for some emerging applications such as real-time traffic monitoring, ecosystem and battlefield surveillance [1]–[4].

Typically, a wireless sensor network is composed of a large number of sensor nodes which are interconnected through wireless links to perform distributed sensing tasks. A sensor node is cheap and with low battery power and computation capacity, but is equipped with sensing, data processing, and communicating components. When sensor nodes sensed some data, they will report their sensing results to a data collection unit (also called *sink*) through a predefined routing. However, since a wireless sensor network is usually deployed at unattended or hostile environments, they are very vulnerable to many security threats such as building bogus routing information, selectively dropping data packet, and creating routing loops to waste the energy of network [5]. In addition, in a wireless sensor network, keeping the source privacy preservation is another great challenge in hostile environments [6]. For example, in a battlefield as shown in Fig. 1, sensor nodes are used to sense the movement of soldiers and report them to the *sink*. If an adversary $\mathcal{A}$ intercepts the sensed data, it can determine the exact location of opposing soldiers. Even though the sensed data is encrypted, the adversary $\mathcal{A}$ can still track the source by traffic analysis [7]. Over the past years, much work has tackled the source privacy preservation issues in wireless sensor networks [6]–[11]. However, most solutions [9]–[11] only apply for a local adversary model, in which an adversary $\mathcal{A}$ has no global view of all network traffic. To date, only few solutions [6]–[8] work in a global adversary model. Therefore, keeping source privacy preservation under global adversary model in wireless sensor networks needs more attention.

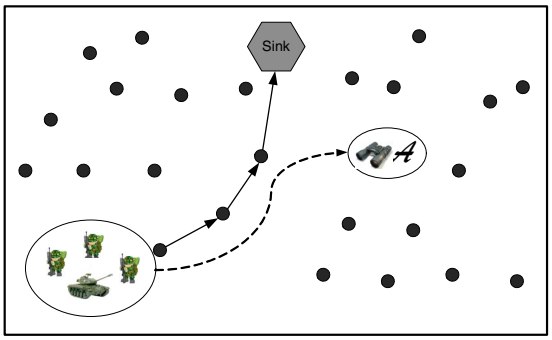

Fig. 1. A wireless sensor network in battlefields

In this paper, we are committed to developing a new Timed Efficient Source Privacy Preservation (TESP^2) scheme against a global adversary who can monitor and analyze all traffic over the whole network. Specifically, based on the topology of wireless sensor networks, we apply the *timed data collection* and *universal re-encryption* [12] techniques to mingle the real sensed data with dummy data for hiding source. The contributions of this paper are threefold.

- First, we use the knowledge on the topology of a wireless sensor network to introduce a new anonymity strategy: *each sensor node (with upstream nodes) will periodically broadcast data collection request to its upstream nodes, and every upstream node will return the real data's ciphertext if it has sensed, or a dummy data's ciphertext if it hasn't. After receiving the real data's ciphertexts from upstream nodes, the sensor node re-encrypts and for-*

978-1-4244-6404-3/10/$26.00 ©2010 IEEE

wards them to its downstream node. With this anonymity strategy, the source of a real data is unobservable, and the source privacy preservation is achieved.

- Second, we develop an efficient filtering technique at each sensor node. With this technique, many dummy data coming from its upstream nodes can be filtered at once. Thus, dummy messages won't swamp the whole network or consume significant energy in the proposed scheme.
- Third, we develop a Java-based simulator to validate the performance of the proposed scheme. The experimental results show that the proposed scheme can achieve high source privacy preservation with some tolerant latency.

The remainder of this paper is organized as follows. In Section II, we review some related work. In Section III, we introduce the system model and design goal. Then, we present the proposed TESP^2 scheme in Section IV, followed by the security analysis and simulation evaluation in Section V and Section VI, respectively. Finally, we draw our conclusions in Section VII.

II. Related Work

Many schemes to keep the source privacy preservation in wireless sensor networks have appeared in the literatures [6]–[11].

In [9], Ozturk et al. use the flooding technique to achieve the source location privacy. In the scheme, the source sends out each sensed data to the *sink* via many paths to make it difficult for an adversary $\mathcal{A}$ to trace the source. However, as pointed in [7], because the *sink* will still receive the sensed data from the shortest path first, the adversary $\mathcal{A}$ can thus quickly trace the source. As a result, the flooding technique consumes a significant amount of energy while providing weak source privacy preservation. In [10], Kamat et al. present two techniques for keeping the source privacy preservation. The first technique is fake packet generation, in which the *sink* creates fake sources whenever a sender has some sensed data to send. The second technique is phantom single-path routing, which achieves location privacy by making every data generated by a source walk a random path before being relayed to the *sink*. In [11], Ouyang et al. use the cyclic entrapment technique to create looping paths at various place in the sensor network, which can cause a local adversary $\mathcal{A}$ to follow these loops repeatedly and achieve the source privacy preservation. Although the above schemes are well designed, they are only efficient for a local adversary model.

To tackle the source privacy preservation issues in a global adversary model, Mehta et al. [7] present two techniques to prevent the leakage of location information: periodic collection and source simulation. Periodic collection provides a high level of location privacy, while source simulation provides tradeoffs between privacy, communication cost, and latency. In [6], Shao et al. propose a scheme called FitProbRate to realize statistically strong source anonymity for sensor networks, in which the policies for dummy traffic generation and for embedding real event messages are discussed.

Different from the aforementioned works, in this paper, based on the topology of wireless sensor networks, we present a new timed efficient source privacy preservation scheme.

III. System Model and Design Goal

In this section, we introduce the network model, the attack model, and identify the design goal.

A. Network Model

We consider a typical wireless sensor network which consists of a *sink* and large numbers of sensor nodes uniformly deployed at a certain interest area, as shown in Fig. 2. The *sink* is a trust and powerful data collection device, which has sufficient computation and storage capabilities and is responsible for collecting the data sensed by sensor nodes. While the sensor nodes are usually low cost devices and stationary at a location to monitor the immediate surroundings.

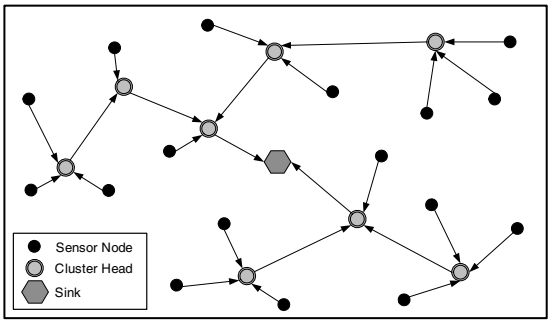

Fig. 2. Wireless sensor networks under consideration

The communication in the network is bidirectional, i.e., two nodes with the wireless transmission range may communicate with each other. Therefore, if a sensor node is close to the *sink*, it can directly contact with the *sink*. However, if a sensor node is far from the transmission range of the *sink*, it should resort to other nodes to establish a route and then communicate with the *sink*. Thus, in a wireless sensor network, some sensor nodes also provide message relay function (MRF) besides the data collection function (DCF). Based on the different functions, we divide all sensor nodes into two types: cluster head nodes and ordinary sensor nodes. The cluster head nodes can support not only MRF but also DCF functions, while the ordinary sensor nodes only provide DCF function. For differentiation purpose, we require each sensor node to have a unique nonzero identifer, i.e., the cluster head nodes are denoted as $\mathcal{C} = \{C_1, C_2, \cdots, C_\alpha\}$ and the ordinary sensor nodes are represented as $\mathcal{O} = \{O_1, O_2, \cdots, O_\beta\}$. For each node $node$ in wireless sensor network, let $\mathbf{USet}(node)$ be the upstream node set of $node$ and $\mathbf{DSet}(node)$ be the downstream node set. Then, we can also differentiate the ordinary sensor nodes, the cluster head nodes and the *sink* in a uniform way. Specifically,

- $node \in \mathcal{O}$, if and only if $\mathbf{USet}(node)$ is empty set and $\mathbf{DSet}(node)$ is nonempty set;
- $node \in \mathcal{C}$, if and only if $\mathbf{USet}(node)$ is nonempty set and $\mathbf{DSet}(node)$ is nonempty set;

- $node$ is the *sink*, if and only if $\mathbf{USet}(node)$ is nonempty set and $\mathbf{DSet}(node)$ is empty set.

In the network model, for each node $node \in \mathcal{O} \cup \mathcal{C}$, the cardinality of $\mathbf{DSet}(node)$ is 1, i.e., $|\mathbf{DSet}(node)| = 1$; for each node $node \in \mathcal{C}$, the cardinality of $\mathbf{USet}(node)$ is more than one, i.e., $|\mathbf{USet}(node)| \geq 1$, so Fig. 2 indicates one topology of such a wireless sensor network.

B. Attack Model

In the attack model, similar as those in [6], we consider an *external*, *global* and *passive* adversary $\mathcal{A}$. Specifically, the adversary $\mathcal{A}$ does not compromise or control any sensor nodes, but has a complete view to eavesdrop and analyze all the communications in the network. Then, with the monitored information, the adversary $\mathcal{A}$ can launch traffic analysis attacks to determine the source of a particular message. Note that the adversary $\mathcal{A}$ could launch some active attacks such as building bogus routing information, selectively dropping true data packet, and creating routing loops to waste the energy of network [5]. However, since the focus of our work is on source privacy preservation, these active attacks are not addressed in this paper.

C. Design Goal

Our goal is to develop a practical *timed data collection* strategy with *universal re-encryption* [12] technique to provide source privacy preservation in wireless sensor networks. Specifically, the following two desirable objectives will be achieved.

- *Achieving the source privacy preservation:* In order to obscure the source of a particular message, many dummy messages are mingled with real sensed data and transmitted to a cluster head C_i simultaneously from $\mathbf{USet}(C_i)$.
- *Reducing the energy cost by filtering dummy messages:* However, if many dummy messages travel over the network, not only the entire network will be swamped but also much energy will be consumed. Therefore, when these dummy messages from $\mathbf{USet}(C_i)$ arrive at the cluster head C_i, they should be filtered by C_i at once.

IV. Proposed TESP^2 Scheme

In this section, we present a new timed efficient source privacy preservation (TESP^2) scheme, which consists of three phases: system initialization phase, sensor nodes deployment phase, and privacy-preservation sensor data report phase. The main idea of the proposed TESP^2 is to resist the traffic analysis attacks by adopting *timed data collection* and *universal re-encryption* [12] technique, which can achieve the source privacy preservation.

A. System Initialization Phase

In system initialization phase, the *sink* first configures the system with TinyECC, a ready-to-use and publicly available library for Elliptic Curve Cryptography in wireless sensor network [13]. Specifically, the *sink* runs the following two steps:

Step 1. Given the security parameter κ, the *sink* first chooses a large prime p such that $|p| = \kappa$. Then, the *sink* chooses two field elements $a, b \in \mathbb{F}_p$ to define the elliptic curve equation $\mathbf{E} : y^2 = x^3 + ax + b \bmod p$ over $\mathbb{F}_p$, where $4a^3 + 27b^2 \neq 0 \bmod p$. In the end, the *sink* chooses a generator point $G = (x_G, y_G)$, whose order q is a larger prime number over $\mathbf{E}(\mathbb{F}_p)$, where G is not the infinite point O.

Step 2. The *sink* chooses a random number $x \in \mathbb{Z}_q^*$ as the master private key, and computes the corresponding public key $Y = xG$. In addition, the *sink* initializes sensor nodes $\mathcal{N} = \{N_0, N_1, N_2, \cdots, N_n\}$ by invoking the Algorithm 1.

Algorithm 1 Sensor Nodes Initialization Algorithm

```
1: procedure SensorNodesInitialization
   Input: un-initialized sensor nodes N = {N_0, N_1, N_2, ..., N_n}
   Output: initialized N = {N_0, N_1, N_2, ..., N_n}
2:     for i = 0 to n do
3:         randomly choose a private key x_i ∈ Z_q^*
4:         compute the corresponding public key Y_i = x_i G
5:         initialize the sensor node N_i and provide N_i with key pair (x_i, Y_i)
6:     end for
7:     return initialized N = {N_0, N_1, N_2, ..., N_n}
8: end procedure
```

B. Sensor Nodes Deployment Phase

In this phase, the *sink* deploys these initialized sensor nodes $\mathcal{N} = \{N_0, N_1, N_2, \cdots, N_n\}$ at a geographical area in various ways such as by air or by land. Given the rich literature in wireless sensor nodes deployment, we do not address the deployment in detail. Without loss of generality, we assume that all sensor nodes will be uniformly distributed in an interest area after deployment. As a result, each sensor node will have multiple immediate neighbors and the neighbors can communicate with each other. To secure the communications, each sensor node $N_i \in \mathcal{N}$ with its neighbor node $N_j \in \mathcal{N}$ will compute the neighbor key $k_{ij} = x_i Y_j = x_j Y_i = x_i x_j G$. Due to the hardness of Computational Diffie-Hellman (CDH) problem over $\mathbf{E}(\mathbb{F}_p)$, each neighbor key k_{ij} is secure against the outside attacks. In the sensor network, if a sensor node N_i only runs the DCF function, it belongs to $\mathcal{O}$. If N_i also runs the MRF function, then it belongs to $\mathcal{C}$. Therefore, as shown in Fig. 2, sensor nodes $\mathcal{N} = \{N_0, N_1, N_2, \cdots, N_n\}$ can be divided into ordinary sensor nodes $\mathcal{O} = \{O_1, O_2, \cdots, O_\beta\}$ and cluster head nodes $\mathcal{C} = \{C_1, C_2, \cdots, C_\alpha\}$ in an *ad hoc* manner.

C. Privacy-Preservation Sensor Data Report Phase

Assume that an ordinary sensor node O_0 senses a data m_0 at time t_0 and is ready to report m_0 to the *sink*, it will run the following steps:

Step 1. O_0 first computes $mac = H(O_0||m_0||t_0||\widetilde{k}_0))$, where $\widetilde{k}_0 = x_0 Y = x_0 x G$ is the static key shared between O_0 and the *sink*, and $H(.)$ is a secure cryptographic hash function. Then, O_0 encodes $O_0||m_0||t_0||mac$ to a point M in $\mathbf{E}(\mathbb{F}_p)$.

Step 2. O_0 chooses two random numbers $k_0, k_1 \in \mathbb{Z}_q^*$ and uses the *universal re-encryption* [12] technique to encrypt M

as $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$, where

$$(\alpha_0, \beta_0, \alpha_1, \beta_1) = (M + k_0 Y, k_0 G, k_1 Y, k_1 G) \quad (1)$$

Step 3. Assume that the cluster head C_1 is the only downstream node in $\mathbf{DSet}(O_0)$. The sensor node O_0 uses its shared key k_{01} to compute $MAC = H(C||k_{01})$. To obscure the originator of C, O_0 doesn't send $C||MAC$ immediately. Instead, O_0 sends $C||MAC$ to C_1 when C_1 launches the data collection request at time t_1.

At time t_1, the cluster head node C_1 broadcasts a data collection request to all nodes in $\mathbf{USet}(C_1)$. Then, each node in $\mathbf{USet}(C_1)$ runs the Algorithm 2 to compute the ciphertext C and sends $C||MAC$ to the cluster head C_1[1].

Algorithm 2 Encrypt sensed data or dummy data

procedure ENCRYPTSENSEDORDUMMYDATA
Input: a sensor node in $\mathbf{USet}(C_1)$
Output: a ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$
  if the sensor node has sensed some data M before time t_1 **then**
    choose two random numbers $k_0, k_1 \in \mathbb{Z}_q^*$
    use the *sink*'s public key $Y = xG$ to compute $C = (\alpha_0, \beta_0, \alpha_1, \beta_1) = (M + k_0 Y, k_0 G, k_1 Y, k_1 G)$
    return the ciphertext C of sensed data M
  else if the sensor node hasn't sensed any data before time t_1 **then**
    choose two random numbers $k_0, k_1 \in \mathbb{Z}_q^*$
    use the cluster head C_1's public key $Y_1 = x_1 G$ to compute $C = (\alpha_0, \beta_0, \alpha_1, \beta_1) = (k_0 Y_1, k_0 G, k_1 Y_1, k_1 G)$
    return the ciphertext C of dummy data O
  end if
end procedure

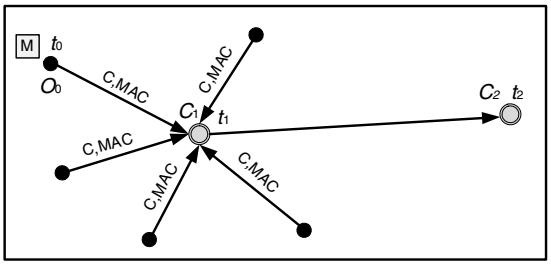

Fig. 3. Timed data collection at the first cluster head

After receiving all messages from its upstream nodes, as shown in Fig. 3, the cluster head node C_1 first checks the validity of each message's MAC, and then further filters the dummy messages by running the Algorithm 3. In the end, the cluster head node C_1 aggregates a group of re-encrypted ciphertexts. To keep the source privacy preversation, C_1 also doesn't immediately send them to its downstream node C_2 until the downstream node C_2 broadcasts a data collection request at time t_2. Without loss of generality, we assume all following cluster head nodes C_i, $i \geq 2$, run the same *timed data collection* as C_1. Then, the sensed data M will eventually arrive at the *sink* at time t_{n+1} by traveling along the path $O_0 \rightarrow C_1 \rightarrow \cdots \rightarrow C_n \rightarrow Sink$, as shown in Fig. 4.

[1]Note that if all neighbor nodes start to send messages at the same time, the probability of collisions is quite high. Therefore, we assume an efficient collision avoidance MAC protocol is employed in the lower layer [14].

Algorithm 3 Filter and re-encryption

procedure FILTERANDREENCRYPTION
Input: ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ sent from an upstream node
Output: $\perp$ or a re-encrypted ciphertext C
  the cluster node C_1 uses its private key x_1 to compute M_0 and M_1, where $M_0 = \alpha_0 - x_1 \beta_0$ and $M_1 = \alpha_1 - x_1 \beta_1$
  if $M_0 = O$ and $M_1 = O$ **then**
    C is a dummy message and filtered
    return $\perp$
  else if $M_0 \neq O$ or $M_1 \neq O$ **then**
    C is a valid message to be sent to the *sink*
    choose two random numbers $k_0', k_1' \in \mathbb{Z}_q^*$ to re-encrypt C as $C = (\alpha_0 + k_0' \alpha_1, \beta_0 + k_0' \beta_1, k_1' \alpha_1, k_1' \beta_1)$
    return the ciphertext C
  end if
end procedure

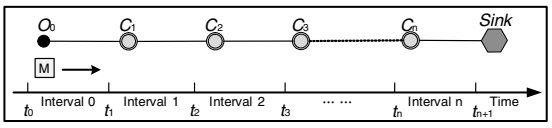

Fig. 4. Timed message relay from the sensor node to the sink

When the ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ arrives at the *sink*, the *sink* can use its private key x to recover m by running the following steps.

Step 1. The *sink* first computes M_0 and M_1, where $M_0 = \alpha_0 - x\beta_0$ and $M_1 = \alpha_1 - x\beta_1$. If $M_0 \neq O$ and $M_1 = O$, the *sink* moves to Step 2; and terminates otherwise.

Step 2. The *sink* decodes M_0 as $O_0||m_0||t_0||mac$, computes the shared key $\widetilde{k}_0 = xY_0 = xx_0 G$ based on the identifier O_0, and then checks $mac \stackrel{?}{=} H(O_0||m_0||t_0||\widetilde{k}_0)$. If it does hold, the sensed data m_0 is accepted. Otherwise, m_0 is rejected.

Correction. When the ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1) = (M + k_0 Y, k_0 G, k_1 Y, k_1 G)$ arrives at C_1, it has the following form

$$\begin{cases} \alpha_0 = M + k_0 Y + k_0' k_1 Y = M + (k_0 + k_0' k_1) Y \\ \beta_0 = k_0 G + k_1' k_1 G = (k_0 + k_0' k_1) G \\ \alpha_1 = k_1' k_1 Y \\ \beta_1 = k_1' k_1 G \end{cases} \quad (2)$$

Clearly, C is still a valid ciphertext at this moment. Therefore,

$$M_0 = \alpha_0 - x\beta_0 = M + (k_0 + k_0' k_1) Y - x(k_0 + k_0' k_1) G = M$$

$$M_1 = k_1 Y - x k_1 G = O$$

Similarly, when the ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ reaches at the *sink*, it is also valid. Thus, the *sink* can successfully recover M from C. As a result, the correctness of TESP^2 follows. More details on the correction of re-encryption technique can be found in [12].

V. SECURITY ANALYSIS

In this section, we discuss security issues in regard to the proposed TESP^2 scheme, i.e., the source privacy protection against the traffic analysis attacks in wireless sensor networks.

• *The re-encryption ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ in the proposed TESP^2 scheme is semantic secure.* Clearly, the

re-encryption ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1) = (M + k_0 Y, k_0 G, k_1 Y, k_1 G)$ in the proposed TESP^2 scheme is the elliptic curve analogy of the universal re-encryption in [12]. Based on the hardness of Decisional Diffie-Hellman (DDH) problem in $\mathbf{E}(\mathbb{F}_p)$, we first prove that $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ is semantic secure under the chosen plaintext attack, which serves as one necessary condition for source privacy preservation.

Let $x, y \in \mathbb{Z}_q^*$, $\widetilde{b} \in \{0, 1\}$. If $\widetilde{b} = 0$, set $T = xyG$. If $\widetilde{b} = 1$, set T to be a random point in $\mathbf{E}(\mathbb{F}_p)$. Given (G, xG, yG, T), the DDH problem is to guess $\widetilde{b}$. Assume that there is an adversary $\mathcal{A}$ which runs in polynomial time and has a non-negligible advantage ε to break the semantic security of C in the proposed TESP^2 scheme, then we can construct another adversary $\mathcal{B}$ which has access to $\mathcal{A}$ and achieves a non-negligible advantage to break the DDH problem.

First, $\mathcal{B}$ is given an DDH instantiation (G, xG, yG, T) as input, where $T = xyG$ when $\widetilde{b} = 0$. $\mathcal{B}$ sets xG as the public key of the *sink* and provides (G, xG) to $\mathcal{A}$. After receiving (G, xG), $\mathcal{A}$ chooses two messages M_0 and M_1 in $\mathbf{E}(\mathbb{F}_p)$ and sends them to $\mathcal{B}$. At this moment, $\mathcal{B}$ plays the role of the Re-encryption challenger, so it flips bit $b \in \{0, 1\}$ and generates ciphertext $C = (\alpha_0, \beta_0, \alpha_1, \beta_1) = (M_b + T, yG, k_1 xG, k_1 G)$, where k_1 is a random number in $\mathbb{Z}_q^*$. In the end, $\mathcal{B}$ sends $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ to $\mathcal{A}$. After receiving $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$, $\mathcal{A}$ returns $\mathcal{B}$ bit b' as the guess of b. $\mathcal{B}$ then guesses $\widetilde{b} = 0$ if $b' = b$.

If $\widetilde{b} = 0$, i.e., $T = xyG$, $\alpha_0 = M_b + xyG$ in C is a valid ciphertext. In this case, A will guess b correctly with probability $\frac{1}{2} + \varepsilon$. Thus, $\Pr[\mathcal{B} \text{ success} | \widetilde{b} = 0] = \frac{1}{2} + \varepsilon$. If $\widetilde{b} = 1$, $\alpha_0 = M_b + T$ is independent with b due to the randomness of T. Therefore, $\Pr[\mathcal{B} \text{ success} | \widetilde{b} = 1] = \frac{1}{2}$. Summarizing the above two cases, we have $\Pr[\mathcal{B} \text{ success}] = \frac{1}{2} \cdot (\frac{1}{2} + \varepsilon) + \frac{1}{2} \cdot \frac{1}{2} = \frac{1+\varepsilon}{2}$. Since ε is assumed non-negligible, the result contradicts with the assumption that DDH is hard in $\mathbf{E}(\mathbb{F}_p)$. Therefore, $C = (\alpha_0, \beta_0, \alpha_1, \beta_1)$ in the proposed TESP^2 scheme is semantic secure.

• *Timed data collection achieves the source privacy preservation.* In the proposed TESP^2 scheme, when a cluster head node C_i broadcasts a data collection request, all nodes in $\mathbf{USet}(C_i)$ will return a message, either a sensed data or a dummy data (if no data is sensed at that time). At the same time, since the re-encryption technique is adopted, each cluster head node actually serves as a mix node, and $\mathbf{USet}(C_i)$ becomes the *anonymity set*, the set of messages which may be linked to an interested item [15]. The larger the *anonymity set*, the more a message can achieve its source privacy, i.e., an adversary $\mathcal{A}$ could identify the real source from a large potential source set. Therefore, TESP^2 is resistant to the traffic analysis attacks and achieves the source privacy preservation. Specifically, with TESP^2, a sensor node, which is n-hop far away from the *sink*, can achieve the *anonymity set* of size $\sum_1^n |\mathbf{USet}(C_i)|$, as shown in Fig. 5. In addition, in the timed message relay, we notice an exceptional feature in TESP^2, i.e., although the dummy messages are involved to achieve the source privacy preservation, they can be filtered by one hop. Thus, the bandwidths in the proposed TESP^2 scheme can be saved.

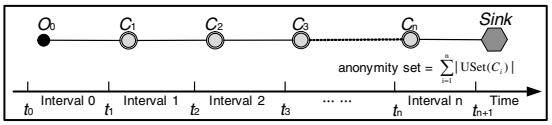

Fig. 5. The size of anonymity set in the timed message relay

VI. Simulation

In this section, we evaluate the average-case performance of the proposed TESP^2 scheme, using a simulator built in Java. The performance metrics used in the evaluation are the size of anonymity set, the average delay for reporting a sensed data, and the energy cost consumed in the timed data collection.

A. Simulation Settings

In the simulations, total N sensor nodes with a transmission radius of 50 meters are first uniformly deployed in an interest area of 1000 m $\times$ 1000 m. Each cluster head node in the area will periodically broadcast the data collection request, and the request time follows the exponentiation distribution with an average rate λ. The detailed parameters settings in the simulations are summarized in Table I. We run the experiments with the different parameters. For each case, 100 networks are randomly generated, and the average performance about the proposed TESP^2 scheme over all of these randomly sampled networks is reported.

TABLE I
Simulation Settings

Parameter	Value
Area	1000 m $\times$ 1000 m
Transmission range	50 m
Number of sensor nodes	$N = [5000, 6000, 7000]$
Data collection request rate	$\lambda = [1/20$ s $, 1/40$ s $, 1/60$ s $]$

B. Simulation Results

Fig. 6 shows the size of *anonymity set* in terms of different hop number varying from 1 to 16 in increment of 1. From the figure, we can see, the larger the hop number, the larger the anonymity set. Since the large-size anonymity set captures high anonymity, the sensor node which is far from the *sink* can achieve better source privacy preservation. On the other hand, from the figure, when the number of sensor nodes N increases, the *anonymity set* also increases. Therefore, the results indicate the high source privacy preservation in the proposed TESP^2 scheme.

Fig. 7 shows the average delay with different parameter λ in terms of different hop number varying from 1 to 16 in increment of 1. From the figure, we can observe i) the average delay increases with the number of hop; ii) the smaller the parameter λ, the longer the delay. Therefore, in order to

reduce the delay, the larger parameter λ is preferred. Assume that sending/receiving one message within one hop requires one unit energy in wireless sensor network, we compare the energy costs under different parameter λ with total 60-minute simulation time in Fig. 8. Clearly, as shown in Fig. 8, the larger the parameter λ, the higher the energy costs. Therefore, there is a tradeoff between the delay and the energy cost when we choose a proper parameter λ in TESP^2.

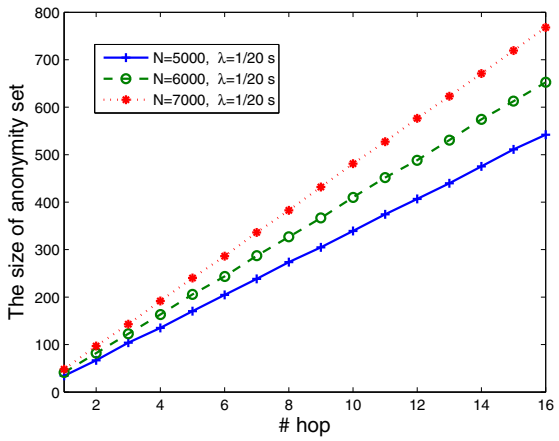

Fig. 6. The anonymity set varies with the number of hops, under different N and $\lambda = 1/20$ s.

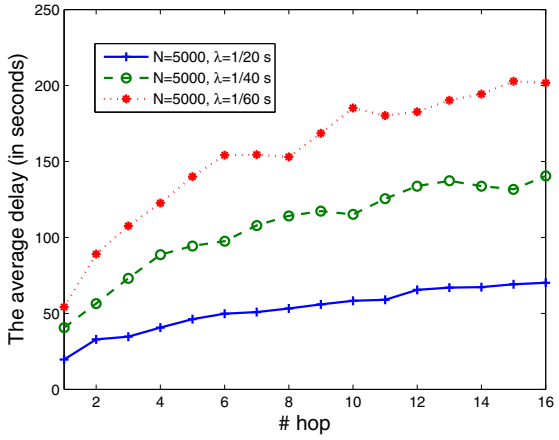

Fig. 7. The average delay varies with the number of hops, under $N = 5000$ and different $\lambda = 1/20\ s, 1/40\ s, 1/60\ s$

VII. Conclusions

In this paper, we have proposed a new timed efficient source privacy preservation (TESP^2) scheme for wireless sensor networks. By adopting the *timed data collection* and *universal re-encryption* technique, the proposed TESP^2 scheme can not only achieve high source privacy preservation but also *early* filter the dummy data to reduce the energy costs. Detailed security analysis has shown that the proposed TESP^2 scheme can resist the traffic analysis attack. In addition, extensive simulations also demonstrate its effectiveness. In our future work, under the conditions that 1) $|\mathbf{DSet}(node)| \geq 1$; and 2)

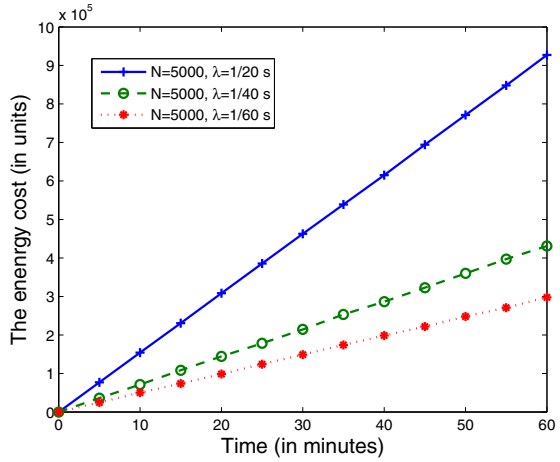

Fig. 8. The energy cost varies with the time in the whole wireless sensor network

an adversary $\mathcal{A}$ can also compromise some sensor nodes for tracking the source, we are ready to generalize the proposed TESP^2 scheme.

References

[1] Akyildiz I, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. *IEEE Communications Magazine* 2002; **40**(8): 102-116.

[2] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring", *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications* (WSNA), Atlanta, Georgia, 2002, pp. 88-97.

[3] R. Lu, X. Lin, C. Zhang, H. Zhu, PH. Ho, and X. Shen, "AICN: an efficient algorithm to identify compromised nodes in wireless sensor network", *International Conference on Communications* (ICC 2008), Beijing, China, May 2008, pp. 1499-1504.

[4] X. Lin, R. Lu, and X. Shen, "MDPA: Multidimensional Privacy-Preserving Aggregation Scheme for Wireless Sensor Networks", *Wireless Communications and Mobile Computing* (Wiley), to appear.

[5] V. C. Giruka, M. Singhal, J. Royalty and S. Varanasi, "Security in wireless sensor networks", *Wireless Communications and Mobile Computing*, Vol. 8, No. 1, pp. 1-24, 2008.

[6] M. Shao, Y. Yang, S. Zhu, and G. Cao, "Towards statistically strong source anonymity for sensor networks", in *INFOCOM 2008*, Phoenix, Arizona, USA, April 15-17, 2008.

[7] K. Mehta, D. Liu, and M. Wright, "Location privacy in sensor networks against a global eavesdropper", in *ICNP 2007*, pp. 314-323, 2007.

[8] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, "Towards event source unobservability with minimum network traffic in sensor networks", in *WiSec 2008*, 2008.

[9] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy constrainted sensor network routing," in *SASN'04*, pp. 88-93, 2004.

[10] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source location privacy in sensor network routing," in *ICDCS'05*, 2005.

[11] Y. Ouyang, Z. Le, G. Chen, J. Ford, and F. Makedon, "Entrapping adversaries for source protection in sensor networks," in *WoWMoM'06*, pp. 23-34, 2006.

[12] P. Golle, M. Jakobsson, A. Juels, and P. Syverson, "Universal re-encryption for mixnets," in *CT-RSA'04*, pp. 163-178, 2004.

[13] A. Liu and P. Ning, "TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks," in *IPSN 2008*, SPOTS Track, pp. 245-256, April 2008.

[14] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: a survey", *IEEE Communications Magazine*, Vol. 44, No. 4, pp. 115-121, 2006.

[15] D. Kelly, R. Raines, and M. Grimaila, "A survey of state-of-the-art in anonymity metrics," in *NDA 2008*, pp. 31-39, October 2008.