symmetry

MDPI

Article

An Efficient Source Location Privacy Protection Algorithm Based on Circular Trap for Wireless Sensor Networks

Yiting Wang [1], Liang Liu [1,*] and Wenzhao Gao [2]

[1] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; eda@nuaa.edu.cn

[2] Department of Computer Science, The University of Hong Kong, Hong Kong 999077, China; eligaowz@hku.hk

* Correspondence: liangliu@nuaa.edu.cn; Tel.: +86-136-4516-0637

Received: 3 March 2019; Accepted: 2 May 2019; Published: 5 May 2019



Abstract: With the aim of addressing the problem of high overhead and unstable performance of the existing Source Location Privacy (SLP) protection algorithms, this paper proposes an efficient algorithm based on Circular Trap (CT) which integrates the routing layer and MAC layer protocol to provide SLP protection for WSNs. This algorithm allocates time slots for each node in the network by using Time Division Multiple Access (TDMA) technology, so that data loss caused by channel collisions can be avoided. At the same time, a circular trap route is formed to induce an attacker to first detect the packets from the nodes on the circular route, thereby moving away from the real route and protecting the SLP. The experimental results demonstrate that the CT algorithm can prevent the attacker from tracking the source location by 20% to 50% compared to the existing cross-layer SLP-aware algorithm.

Keywords: wireless sensor network; source location privacy; energy efficiency; TDMA

1. Introduction

Wireless sensor network (WSN) is a multihop self-organizing network composed of multiple sensor nodes. Due to its small size of nodes, strong networking capability, and convenient deployment and maintenance, it is widely deployed for environmental monitoring [1–4], intelligent transportation [5–8], military defense [9,10] and medicine and health [11,12]. In WSN applications for asset (e.g., rare animal) monitoring, the location information is routed from the source node that detects the asset to the base station. Attackers can find out the source node (asset) location by using a hop-by-hop backtracking strategy for illegal purposes. Therefore, how to ensure the privacy of the source location is a problem that urgently needs to be solved.

The existing source location privacy (SLP) protection methods [13–18] mainly protect the SLP by increasing the variety of routing paths, such as phantom routing, multi-path routing, and false information source injection and other secure routing mechanisms. Most of these technologies sacrifice network performance for a certain level of security and are not suitable for applications that require high query efficiency. Additionally, the existing MAC-layer SLP protection algorithm SLP-TDMA-DAS (STD) [19] protects the SLP by generating a decoy single-path and adjusting the priority of its broadcast slot to interfere with the attacker's hop-by-hop traceback. However, if the listening range does not cover the decoy path, no matter how high the time slot priority of the nodes on the decoy path, it will not attract the attacker to move to that path.

In order to solve these problems, this paper proposes an efficient algorithm based on Circular Trap (CT) which integrates the routing layer and MAC layer protocol to provide SLP protection. This algorithm constructs a circular route using the geographic routing protocol and adjusts the time slot of its nodes to be lower than that of its neighbors. When an attacker eavesdrops on a network, it will first receive messages from the circular route and circulate through it. If the attacker starts from the base station to track the source node location, it is inevitable for the attacker to receive the message broadcast by the circular trap path, thus extending the capture time of the attacker. If an attacker listens in a random location in the network, the circular trap can also increase the probability of the attacker being disturbed. Compared with the STD algorithm, the CT algorithm has higher stability and security. We make the following novel contributions: (i) we introduce the concept of Circular Trap upon TDMA to protect SLP; (ii)we propose a 3-phase distributed algorithm using the Circular Trap method. (iii) we propose a routing protocol to adjust the time slots of the nodes when constructing a circular route.

The rest of this paper is organized as follows: We summarize the related work in Section 2 and introduce the preliminaries in Section 3. Section 4 illustrates the basic idea of CT algorithm. The evaluation of CT algorithm is conducted in Section 5. Finally, we conclude this paper in Section 6.

2. Related Work

The protection of SLP in sensor networks has always been an important research hotspot. The SLP problem was first introduced in references [13,14] as a panda hunter game. The Save-The-Panda Organization submits a query for looking up a panda, the nodes in the network collect and return relevant location information to the base station when the target panda is detected. In the game, there is a Panda-Hunter (attacker) who is attempting to find out the asset by using a hop-by-hop backtracking strategy.

Up to now, many schemes have been proposed to provide the SLP against eavesdroppers [20]. Ozturk et al. [13] propose an SLP protection protocol based on random walk, which is called Phantom routing. The protocol is divided into two phases: (1) the data packet first randomly walks h hops to reach a pseudo-source node; (2) the data packet arrives at the sink from the pseudo source node through flooding or single-path routing. Li et al. [21] propose a novel routing strategy to protect the SLP, which consists of directional random routing, h-hop routing in the annular region and the shortest path routing. In the Reference [16], the authors propose a SLP protection strategy via pseudo normal distribution-based phantom routing. Zhou et al. [22] protect the SLP by utilizing the sensor ability of perceiving the presence of a mobile attacker nearby.

There are fewer techniques using a cross-layer approach to provide SLP. In reference [23], the beacon frames used by the MAC layer are modified to propagate messages away from the source before convergence. Kirton et al. [19] propose the MAC layer SLP protection algorithm STD, which protects the SLP by generating a decoy single-path and adjusting the priority of its broadcast slot to interfere with the attacker's monitoring. However, as shown in Figure 1, if the listening range of attackers does not cover the decoy path, no matter how high the time slot priority of the nodes on the decoy path is, it will not encourage the attacker to move to it.

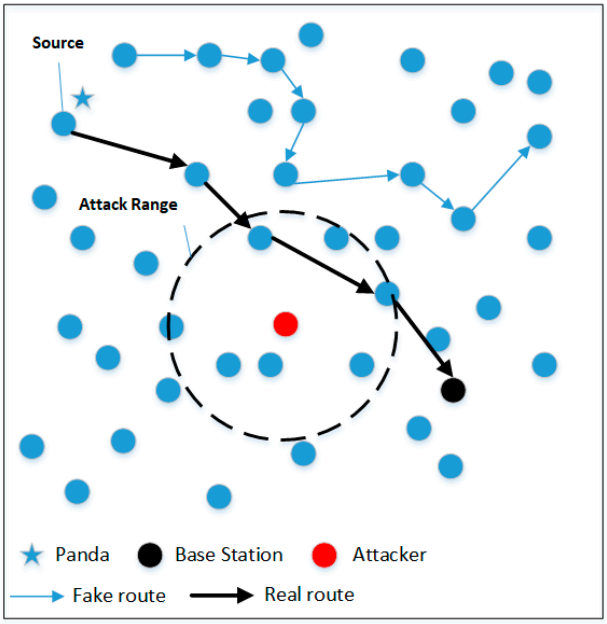


Figure 1. The problem of STD algorithm.

3. Preliminaries

3.1. Network Model

The network model of this work is similar to that of the panda hunter game [13]. In a homogeneous sensor network $\mathcal{W}$, there are N sensor nodes, and $\mathcal{W} = \{V_i \mid 1 \leq i \leq N\}$. In this network, each sensor node has the same computing performance, storage space and battery energy, and each node V_i knows the fixed position of Sink (x_s, y_s) and its own position (x_i, y_i) by GPS or location algorithm. When a node V_t detects an asset, it becomes the source node and notifies the sink node about the information of the asset. Assume that the sensor network is deployed in an unobstructed planar space. The distance between sensor nodes is Euclidean distance. If the positions of nodes V_i and V_i are (x_1, y_1) and (x_2, y_2) respectively, the distance between the two points is:

$$\mathrm{d}(V_1,\ V_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}, \tag{1}$$

In a sparse network, it is easier for an attacker to locate the direct sender and receiver of a packet due to the sparse number of neighbor nodes. Therefore, this paper assumes that the network is densely connected. In terms of graph theory, we represent WSN as undirected graph $G = (V, E)$, where vertex set V represents nodes and edge set E represents communication links between node pairs.

3.2. Attack Model

In sensor networks, data routing uses hop-by-hop transmission because of the limited communication range of each sensor node. The attacker can trace the data to the sink or source node according to the time correlation of data packet transmission and the traffic pattern of different communication nodes. We assume that the attacker has the following characteristics: (1) local, that is, the attacker's surveillance range is its neighboring sensor node; (2) passive, that is, the attacker can only listen for packets in the network and cannot control or destroy sensor nodes, it will not have any functional impact on the network; (3) mobile, the attacker will start from different locations in the network to find the location of the source node.

In a typical SLP protection problem, there are two main types of attackers: (1) The patient attacker moves to the direction in which the packet is sent when new data is captured, otherwise it remains in the in-place wait state; (2) The curious attacker will walk randomly if it does not receive any packet

within the limited waiting time. Curious attackers are more flexible when they cannot receive any data. Therefore, in this paper, we implement the attacker model [19] based on a curious attacker, as shown in Algorithm 1.

Algorithm 1 Attacker model.

input: R, H, M, s_0, D
output: *NULL*
1. $msgs$, $moves$, num, $curLoc \leftarrow \emptyset$, 0, 0, s_0;
2. $history[] \leftarrow$ array of node ids of size H;
3. **IF** $msgs.size() < R$ **THEN**
4. $Msgs \leftarrow msgs \cup \{l\}$;
5. **END IF**
6. **IF** ($msgs \neq \emptyset$)
7. **IF** ($moves < M$)
8. **IF** ($H > 0$)
9. $history[num] \leftarrow curLoc$;
10. $num \leftarrow (num + 1)\ \%\ H$;
11. **END IF**
12. $curLoc \leftarrow$ D($msgs$, $history$);
13. $moves \leftarrow moves + 1$;
14. $msgs \leftarrow \emptyset$;
15. **END IF**
16. **END IF**

3.3. Capture Time

In order to evaluate whether a solution provides SLP protection services, the concept of Security Time [14] was proposed, which is first used to evaluate the parameters of how many messages an attacker needs to trace to the source node. But the original concept raises some questions, because the number can be very large or infinite (and can never be traced), making it difficult to evaluate the performance of the solution. A similar concept is used in this paper to evaluate the performance of an algorithm called Capture Time [19], which describes the minimum time required for an attacker to trace the source node location in a given network.

In the CT algorithm, for the same network $G = (V, E)$, the capture time under normal DAS (Data Aggregation Scheduling) is T_D, and in the case of scheduling algorithm S, it is T_S. In general, $T_S = E \times T_D$ $(1 < E < 2)$, where E is the expected value of the schedule, representing the user's expectation of the effect of the scheduling algorithm S in extending the capture time. In the experiment, if the actual time T used by attacker $\mathcal{A}$ to track the source node is greater than the capture time T_S, it is considered that this scheduling S successfully protects the SLP.

3.4. TDMA (Time Division Multiple Access)

TDMA is a channel access method for shared media networks that allows multiple users to share the same frequency channel by dividing the signal into different slots [24–26]. Time is divided into equal time slices and a time slice is divided into different time slots. Each node can be allocated to different time slots with a different serial number. The node only transmits data when the time reaches its allocated time slot. Conflicts between nodes can be avoided within the same frequency. In the CT algorithm, the smaller the serial number of the time slot is, the higher the time priority of the time slot is. Nodes with smaller serial number of the time slot should send data before those with larger serial number.

3.5. DAS (Data Aggregation Scheduling)

The DAS (data aggregation schedule) is a sequence of sender sets $\{\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_l\}$ satisfying the following conditions:

(1) $\sigma_i \cap \sigma_j = \varnothing$, $\forall i \neq j$

(2) $\bigcup_{i=1}^{l} \sigma_i = V - \{Sink\}$

(3) Data are aggregated from σ_k to $V - \bigcup_{i=1}^{k} \sigma_i$ at time slot k, for all $k = 1, 2, \cdots, l$ and finally all the data are aggregated to the sink in time slot l.

Reference [19] defines DAS as strong DAS and weak DAS. The main difference between these forms is that the strong DAS requires a node to have all of its neighbors closer to the sink with a slot higher than its own, while a weak DAS only requires at least one such neighbor to transmit later.

4. CT Algorithm

4.1. The Idea of CT

As shown in Figure 2, this algorithm uses the geographic routing protocol to construct a circular route and makes the time slot of the node on the circular route smaller than its neighbors through adjusting the time slot, so that when an attacker eavesdrops and tracks in the network, messages from the ring route will be monitored first, and then the attacker will loop through the ring trap.

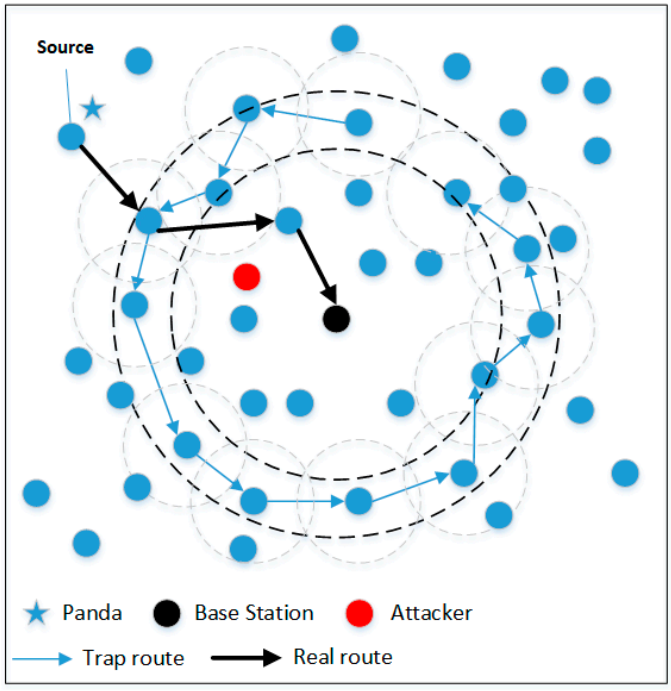


Figure 2. CT algorithm.

4.2. The Design of CT

We divide the CT algorithm into three phases: DAS slot allocation, trap starting point selection and trap construction. The detailed design of each phase is as follows.

4.2.1. Phase 1: DAS Slot Allocation

The algorithm in this phase starts from the base station, and each node stores its potential parent node, the number of hops between the node and the base station, and the allocated time slot information. When a node receives a set of "non-empty" messages, it will select one of the senders as its parent

node, add the remaining nodes as potential parents, and update its own time slot to be less than the time slots of all neighbor nodes until the whole network time slot allocation is completed.

When a node broadcasts its own state information, the information includes its one-hop neighbors, so each node can store the information of its own two-hop neighbors when receiving the state information of its neighbors. Such two-hop node information can cause each node to check whether its own time slot conflicts with the time slot of its two-hop neighbors and be used to resolve the conflict. The process at this phase is shown in Algorithm 2. The algorithm of phase 1 is described as follows:

(1) Initialize the slot of Sink node to the total number of nodes, so that the whole network has sufficient slot allocation and the slots allocated by each node are different.
(2) Nodes in the network continuously broadcast messages until there is no slot conflict between each node and its two-hop neighbor nodes. If the slot of the receiving node has not been determined but the slot of the sending node has been determined, the sending nodes become all potential parent nodes of the receiving node. In addition, among the one-hop neighbor nodes of the sending node, the nodes that are not allocated by the slot may be potential competitors of the receiving node, that is, they may be assigned to the same slot.
(3) Detect slot conflict and update slot. If the slot is updated from parent, and the slot of the current node is greater than or equal to the slot of parent node, the slot value will be updated.

4.2.2. Phase 2: Trap Starting Point Selection

In the protocol of this phase, the algorithm sends search messages in any direction according to the preset trap radius R and selects a node as the starting node of the trap in phase 3. The process of this phase is shown in Algorithm 3. The algorithm of phase 2 is described as follows:

(1) Broadcast the SEARCH message from the Sink node.
(2) Along a certain direction (for example, the Y-axis), select the node with the largest y-coordinate in the neighbors as the next node.
(3) If the distance from the current node to the sink node is the preset trap radius R, select the node as the start node, otherwise repeat step 2.

4.2.3. Phase 3: Trap Construction

The node selected in phase 2 is the starting node of the algorithm of this phase. As shown in Algorithm 4, the steps in this phase are as follows:

(1) the starting node broadcasts *TRAP* message, and the message content is $\langle TRAP, i, Next, nSlot \rangle$, representing trap, sending node id, next target node, and minimum time slot respectively.
(2) after receiving the *TRAP* information, the node selects a node that meets the following conditions as the next target node among all the nodes: (a) the target node is located in the annular region in Figure 3; (b) the target node is within the communication range; (c) $\angle AOi$ has the largest angle, which is calculated from the positions of the current node A (x_a, y_a), its one-hop neighbor I (x_i, y_i) and sink O (x_o, y_o) by cosine theorem; (d) the target node adjusts its own slot value to be the smallest of the neighbor nodes except the nodes on the selected path.
(3) repeat step 2 until all nodes have been selected.

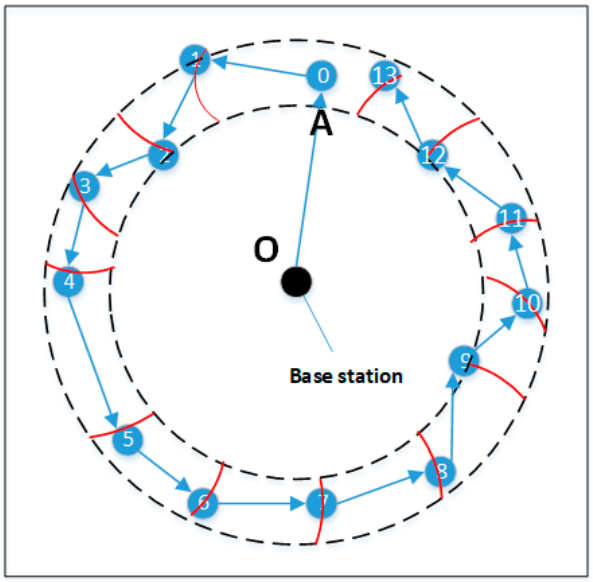


Figure 3. Trap construction.

Algorithm 2: DAS protocol of node *i*

BROADCAST<*DISSEM*, *Normal*, *i*, *myN*, *par*>;
IF (*i* = *Sink*) **THEN**
  hop ← 0;
  par ← 0;
  slot ← *nNodes*;
END IF
IF (*Nor* = *TRUE*) **THEN**
  IF (*slot* = -1 ∧ *j.slot* ≠ -1) **THEN**
    Ppar ← *Ppar* ∪ {*j*};
      Others ← *Others* ∪ {*n* | *myN*[*n*].*slot* = -1};
  END IF
  twohopsNeigh ← *twohopsNeigh* ∪ *N*;
ELSE
  IF (*par* = *j*) **THEN**
    IF (*slot* ≥ *j.slot*) **THEN**
      slot ← *j.slot* - 1;
      Normal ← *FALSE*;
    END IF
  END IF
END IF
IF (*slot* = -1) **THEN**
  hop ← min{*k.hop* | k ∈ *Ppar*} + 1;
  Par ← min{k | k.hop = hop – 1};
  slot ← *par.slot* – rank(*i*, *Others*) – 1;
  par.children ← *par.chidren* ∪ *i*;
END IF
IF (∃ *j*, *j* ≠ *i* ∧ *j* ∈ *twohopsNeigh* ∧ *j.slot* = *slot*) **THEN**
  IF (*hop* > *j.hop* ∨ (*hop* = *j.hop* ∧ *i* > *j*)) **THEN**
    slot ← *slot* – 1;
  END IF
END IF

Algorithm 3: Node location protocol for node i

```
1.  BROADCAST<SEARCH, i, Next, Distance>;
2.  IF (i = Sink) THEN
3.    Next ← {n | n ∈ myN ∧ n.getY() = Max(myN.getY()};
4.    Distance ← Distance – 1;
5.  END IF
6.  IF (n = i) THEN
7.    IF (d = 0) THEN
8.      startNode ← i;
9.    ELSE
10.         Next ← {n | n ∈ myN ∧ n.getY() = Max(myN.getY()};
11.       Distance ← Distance – 1;
12.   END IF
13. END IF
```

Algorithm 4: Circular setting protocol of node i

```
1.  BROADCAST<TRAP, i, Next, nSlot>;
2.  IF (i = start) THEN
3.    nSlot ← min({n.slot | n ∈ myN} ∪ {slot});
4.    Next ← choose({n | n ∈ Circular Area ∧ n ∈ Sensing area of node i ∧ ∠AOn = max(∠AOi | i ∈ nodes)});
5.  END IF
6.  IF (n = i) THEN
7.    slot ← s – 1;
8.    nSlot ← min({n.slot | n ∈ myN} ∪ {slot});
9.    Next ← choose({n | n ∈ Circular Area ∧ n ∈ Sensing area of node i ∧ ∠AOn = max(∠AOi | i ∈ nodes)});
10.   IF (Next ∉ Trap Route) THEN
11.     BROADCAST<TRAP, i, Next, nSlot>;
12.   ELSE
13.     BREAK;
14.   END IF
15. END IF
```

As shown in Figure 3, the order of $\angle AOi$ for each node is as follows: $\angle AO0 < \angle AO1 < \ldots \angle AOi \ldots < \angle AO13$. Therefore, the algorithm runs counterclockwise along the annular region in the figure and gradually adjusts the time slot.

5. Evaluation

In this section, we evaluate the performance of DAS scheduling, STD algorithm and CT algorithm through simulations on the simulator [27].

The experiments are conducted on a PC with an Intel Core i5-4210U(2.40GHz) CPU and 8 GB memory running Ubuntu operating system and the Eclipse IDE. The sensor nodes are uniformly deployed in a rectangle monitor area. The size of this area is $2000 \times 2000\ m^2$. The experiment evaluates the security performance of the algorithm by testing the protection success rate (SR) value of the source node under different parameter settings. $SR = \frac{ST}{NE} \times 100\%$, where ST is the number of times of security, which means the number of times the attacker actually captures time is greater than the expected capture time in the experiment, and NE is the number of experiments, that is, the total number of algorithm tests. According to Section 3.2, $T_S = E \times T_D\ (1 < E < 2)$, where T_S is the attack time of the attacker under the scheduling S, T_D is the attack time of the attacker under the normal DAS scheduling, and E is the expected value of the schedule. We set the value of E to 1.5 in the experiment, and if the actual capture time T is greater than T_S, it is considered that the SLP is successfully protected. The other parameters are shown in Table 1.

Table 1. Experimental parameters.

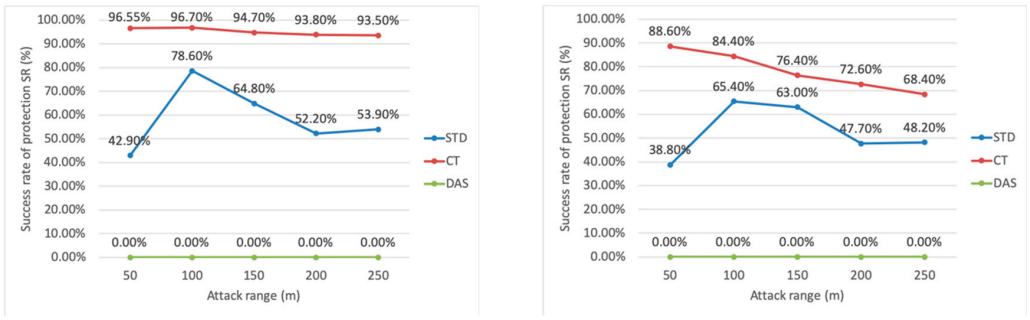
Parameter Name	Parameter Value
Network parameters	
Network coverage area	2000×2000 m^2
Network node distribution	Uniform grid distribution
Node communication radius	50 m
Number of nodes	1600
Number of experiments NE	1000
Attack Range	50
STD algorithm parameters	
Search length d	3
False path length lenD	20
CT algorithm parameters	
Large circle radius of the ring	400 m
Small circle radius of the ring	300 m

Table 2 and Figure 4 shows the performance of STD, DAS, and CT under different attack ranges. As can be seen from the experimental results, in most cases, both STD algorithm and CT algorithm interfere with the listening and tracking behavior of attackers, while the normal DAS scheduler has no impact on them. The STD algorithm is vulnerable to the impact of the size of the attacker's attack range, which makes the privacy protection effect of the algorithm fluctuate. In contrast, the privacy protection effect of the CT algorithm is relatively stable, and the success rate of protection is increased by 20%~50% compared with the STD algorithm. Figure 5 shows the performance of STD, DAS, and CT under different number of nodes. As the number of nodes increases, the number of hops the attacker can trace back to find the source node increases, so we can see from the figure that the success rate of protection increases as the number of nodes increases. Figure 6 shows the performance of CT under a different trap radius. From the figure we can see that when the trap radius is less than 500 m, the attacker is more likely to detect the node on the trap, and the protection success rate increases as the trap radius increases. However, when the trap radius is larger than 500 m, the probability that the attacker and the source node fall into the trap at the same time increases, which causes the trap to be ineffective, and the protection success rate is reduced.

Figure 7 shows the latency of STD, DAS, and CT under 20 different networks. We can know from the experimental results that DAS algorithm has the minimum latency. The CT and STD algorithms adjust the slot values of some nodes to construct trap, so the TDMA slot value of the network increases. In order to improve the security of the algorithm, a slight network latency is allowed within our acceptable range. Figure 8 shows the latency of STD, DAS and CT under different number of nodes. The maximum value of TDMA slot increases as the number of nodes increases, so we can see from the figure that the latency increases as the number of nodes increases. Figure 9 shows that the latency increases as the radius of the trap increases.

Table 2. Experimental results.

Experimental results of STD algorithm under different attack ranges							
AR(m)	TD(s)	E	TS(s)	Number of T>TS	Number of T≤TS	Number of experiments	SR(%)
50	153	1.5	229.5	858	1142	2000	42.90%
100	115	1.5	172.5	1572	428	2000	78.60%
150	101	1.5	151.5	1296	704	2000	64.80%
200	92	1.5	138	1044	956	2000	52.20%
250	87	1.5	130.5	1078	922	2000	53.90%
Experimental results of CT algorithm under different attack ranges							
AR(m)	TD(s)	E	TS(s)	Number of T>TS	Number of T≤TS	Number of experiments	SR(%)
50	153	1.5	229.5	1931	69	2000	96.55%
100	115	1.5	172.5	1934	66	2000	96.70%
150	101	1.5	151.5	1894	106	2000	94.70%
200	92	1.5	138	1876	124	2000	93.80%
250	87	1.5	130.5	1870	130	2000	93.50%
Experimental results of DAS algorithm under different attack ranges							
AR(m)	TD(s)	E	TS(s)	Number of T>TS	Number of T≤TS	Number of experiments	SR(%)
50	153	1.5	229.5	0	2000	2000	0.00%
100	115	1.5	172.5	0	2000	2000	0.00%
150	101	1.5	151.5	0	2000	2000	0.00%
200	92	1.5	138	0	2000	2000	0.00%
250	87	1.5	130.5	0	2000	2000	0.00%



(**a**) (**b**)

Figure 4. The impact of the attack range on the success rate of privacy protection: (**a**) fixed attacker position; (**b**) random attacker location.

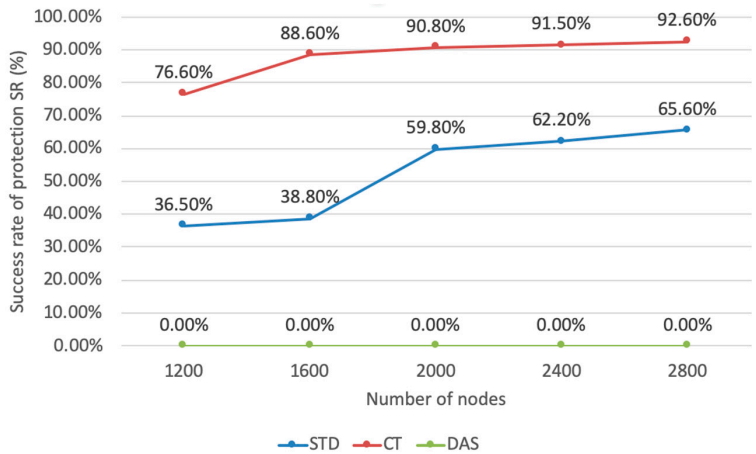


Figure 5. The impact of the number of nodes on the success rate of privacy protection.

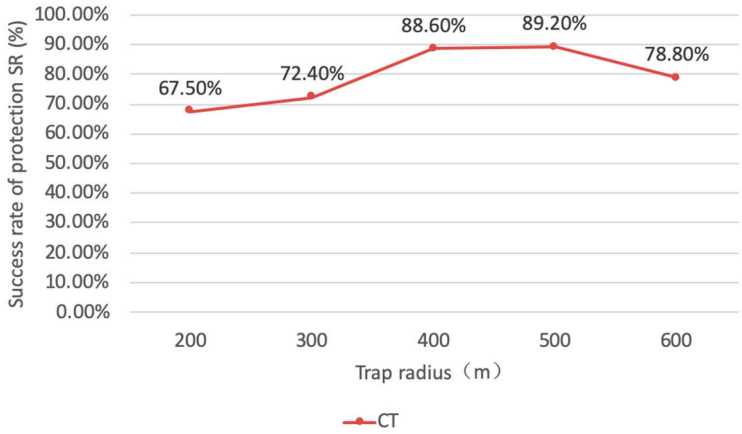


Figure 6. The impact of trap radius on the success rate of privacy protection.

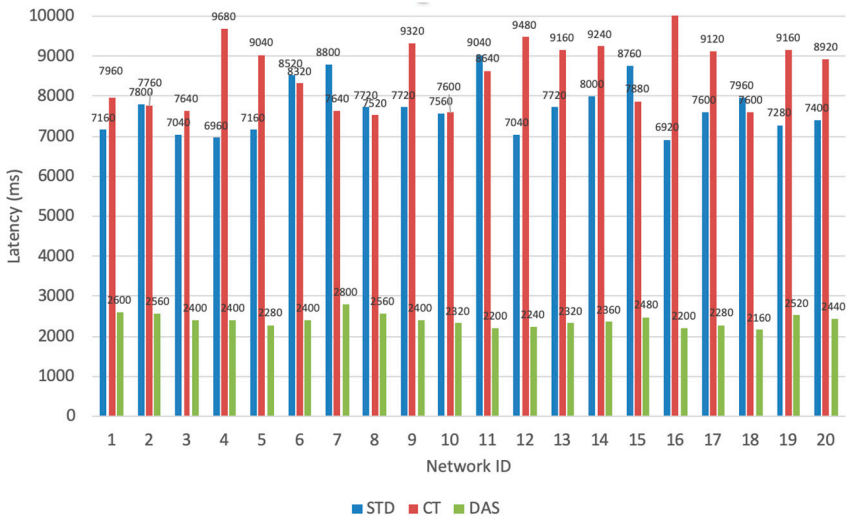


Figure 7. The latency under different network topologies.

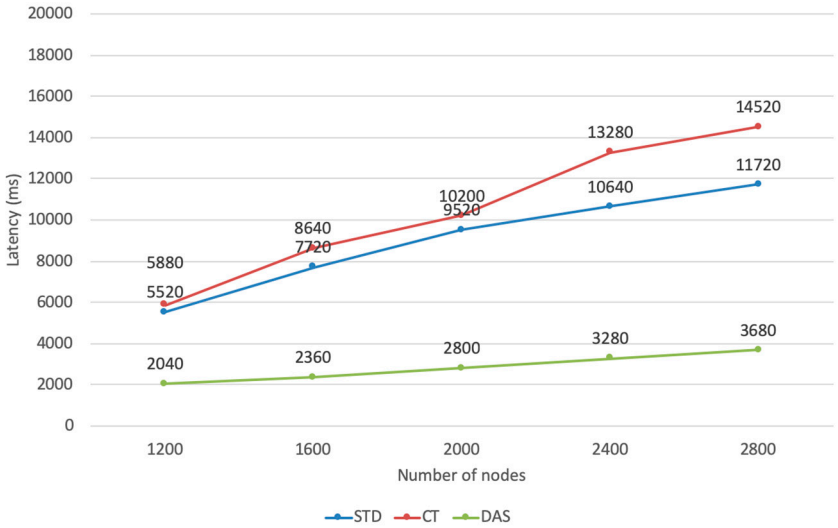


Figure 8. The impact of the number of nodes on latency.

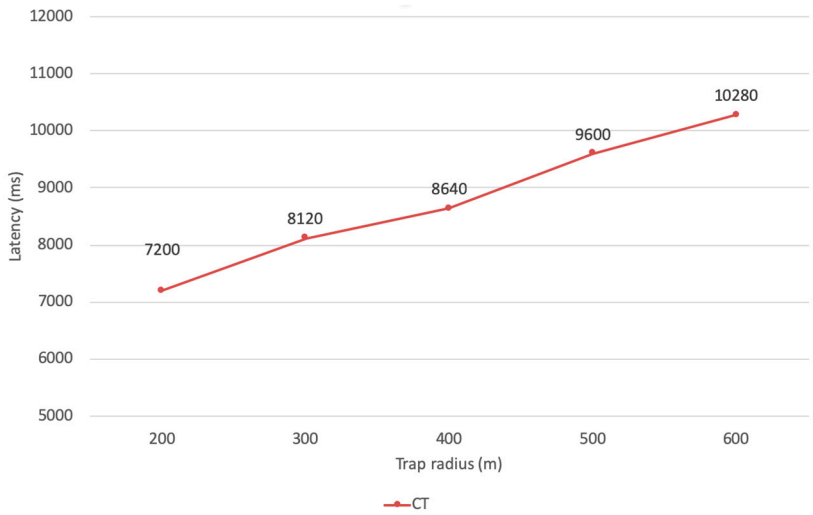


Figure 9. The impact of trap radius on latency.

6. Conclusions

The existing SLP protection algorithms mainly protect the SLP by changing or increasing the routing path length, which greatly increases the energy overhead. To solve this problem, the CT algorithm which integrates the routing layer and MAC layer protocol to provide SLP protection is proposed. The algorithm constructs a circular route using a geographic routing protocol and adjusts the time slots of its nodes to be lower than that of its neighbor nodes. When an attacker eavesdrops on the network, it first receives the message from the circular and circulates through it. The experimental results of the algorithm show that our algorithm is superior to the existing DAS scheduling algorithm and STD algorithm.

Author Contributions: Conceptualization, Y.W. and L.L.; Methodology, Y.W.; Software, W.G.; Validation, L.L.; Formal Analysis, Y.W.; Investigation, L.L.; Data Curation, W.G.; Writing—Original Draft Preparation, Y.W. and W.G.; Writing—Review and Editing, Y.W. and W.G.; Visualization, Y.W.; Supervision, L.L.; Project Administration, Y.W.

Funding: This work is supported by the National Natural Science Foundation of China under Grant No.(61402225, 61373015, 41301407), the National Natural Science Foundation of Jiangsu Province under Grant No.BK20140832, the China Postdoctoral Science Foundation under Grant No.2013M540447, the Jiangsu Postdoctoral Science Foundation under Grant No.1301020C, State Key Laboratory for smart grid protection and operation control Foundation, Science and Technology Funds from National Electric Net Ltd.(The Research on Key Technologies of Distributed Parallel Database Storage and Processing based on Big Data), the Foundation of Graduate Innovation Center in NUAA under Grant No.kfjj20181608.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Szewczyk, R.; Mainwaring, A.; Polastre, J.; Anderson, J.; Culler, D. An analysis of a large scale habitat monitoring application. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MD, USA, 3–5 November 2004; pp. 214–226.
2. Lazarescu, M.T. Design of a WSN platform for long-term environmental monitoring for IoT applications. *IEEE J. Emerg. Sel. Top. Circuit. Syst.* **2013**, *3*, 45–54. [CrossRef]
3. Tellez, M.; El-Tawab, S.; Heydari, H.M. Improving the security of wireless sensor networks in an IoT environmental monitoring system. In Proceedings of the 2016 IEEE Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA, 29 April 2016; pp. 72–77.
4. Yang, J.; Zhou, J.; Lv, Z.; Wei, W.; Song, H. A real-time monitoring system of industry carbon monoxide based on wireless sensor networks. *Sensors* **2015**, *15*, 29535–29546. [CrossRef] [PubMed]
5. Lee, U.; Zhou, B.; Gerla, M.; Magistretti, E.; Bellavista, P.; Corradi, A. Mobeyes: Smart mobs for urban monitoring with a vehicular sensor network. *IEEE Wirel. Commun.* **2006**, *13*, 52–57. [CrossRef]

6. Sherly, J.; Somasundareswari, D. Internet of things based smart transportation systems. *Int. Res. J. Eng. Technol.* **2015**, *2*, 1207–1210.
7. Jo, Y.; Jung, I. Analysis of vehicle detection with WSN-based ultrasonic sensors. *Sensors* **2014**, *14*, 14050–14069. [CrossRef] [PubMed]
8. Srivastava, J.R.; Sudarshan, T. Intelligent traffic management with wireless sensor networks. In Proceedings of the 2013 ACS International Conference on Computer Systems and Applications (AICCSA), Ifrane, Morocco, 27–30 May 2013; pp. 1–4.
9. Butun, I.; Morgera, S.D.; Sankar, R. A survey of intrusion detection systems in wireless sensor networks. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 266–282. [CrossRef]
10. Furtak, J.; Zieliński, Z.; Chudzikiewicz, J. Security techniques for the WSN link layer within military IoT. In Proceedings of the 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), Reston, VA, USA, 12–14 December 2016; pp. 233–238.
11. Stankovic, J.; Cao, Q.; Doan, T.; Fang, L.; He, Z.; Kiran, R.; Lin, S.; Son, S.; Stoleru, R.; Wood, A. Wireless sensor networks for in-home healthcare: Potential and challenges. In Proceedings of the High Confidence Medical Device Software and Systems (HCMDSS) Workshop, Philadelphia, PA, USA, 2–3 June 2005.
12. Adame, T.; Bel, A.; Carreras, A.; Melià-Seguí, J.; Oliver, M.; Pous, R. CUIDATS: An RFID–WSN hybrid monitoring system for smart health care environments. *Future Gener. Comput. Syst.* **2018**, *78*, 602–615. [CrossRef]
13. Ozturk, C.; Zhang, Y.; Trappe, W. Source-location privacy in energy-constrained sensor network routing. In Proceedings of the 2nd ACM workshop on Security of ad hoc and Sensor Networks, Washington, DC, USA, 25 October 2004; pp. 88–93.
14. Kamat, P.; Zhang, Y.; Trappe, W.; Ozturk, C. Enhancing source-location privacy in sensor network routing. In Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05), Columbus, OH, USA, 6–10 June 2005; pp. 599–608.
15. Lightfoot, L.; Li, Y.; Ren, J. Preserving source-location privacy in wireless sensor network using STaR routing. In Proceedings of the 2010 IEEE Global Telecommunications Conference GLOBECOM 2010, Miami, FL, USA, 6–10 December 2010; pp. 1–5.
16. Huang, J.; Sun, M.; Zhu, S.; Sun, Y.; Xing, C.C.; Duan, Q. A source-location privacy protection strategy via pseudo normal distribution-based phantom routing in WSNs. In Proceedings of the 30th Annual ACM Symposium on Applied Computing, Salamanca, Spain, 13–17 April 2015; pp. 688–694.
17. Mehta, K.; Liu, D.; Wright, M. Protecting location privacy in sensor networks against a global eavesdropper. *IEEE Trans. Mob. Comput.* **2012**, *11*, 320–336. [CrossRef]
18. Yang, Y.; Shao, M.; Zhu, S.; Urgaonkar, B.; Cao, G. Towards event source unobservability with minimum network traffic in sensor networks. In Proceedings of the First ACM Conference on Wireless Network Security, Alexandria, VA, USA, 31 March–2 April 2008; pp. 77–88.
19. Kirton, J.; Bradbury, M.; Jhumka, A. Source location privacy-aware data aggregation scheduling for wireless sensor networks. In Proceedings of the 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, USA, 5–8 June 2017; pp. 2200–2205.
20. Conti, M.; Willemsen, J.; Crispo, B. Providing source location privacy in wireless sensor networks: A survey. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 1238–1280. [CrossRef]
21. Li, S.; Xiao, Y.; Lin, Q.; Qi, Z. A novel routing strategy to provide source location privacy in wireless sensor networks. *Wuhan Univ. J. Nat. Sci.* **2016**, *21*, 298–306. [CrossRef]
22. Qian, Z.; Xiaolin, Q.; Youwei, D. Intelligent Silent Zone for Source-Location Privacy Based on Context-Awareness in WSNs. *Trans. Nanjing Univ. Aeronaut. Astronaut.* **2018**, *23*, 203–218.
23. Shao, M.; Hu, W.; Zhu, S.; Cao, G.; Krishnamurth, S.; La Porta, T. Cross-layer enhanced source location privacy in sensor networks. In Proceedings of the 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, Rome, Italy, 22–26 June 2009; pp. 1–9.
24. Cionca, V.; Newe, T.; Dadârlat, V. TDMA protocol requirements for wireless sensor networks. In Proceedings of the 2008 Second International Conference on Sensor Technologies and Applications (sensorcomm 2008), Cap Esterel, France, 25–31 August 2008; pp. 30–35.
25. Bai, R.G.; Qu, Y.G.; Guo, Y.; Zhao, B.H. An energy-efficient TDMA MAC for wireless sensor networks. In Proceedings of the 2nd IEEE Asia-Pacific Service Computing Conference (APSCC 2007), Tsukuba Science City, Japan, 11–14 December 2007; pp. 69–74.

26. Alvi, A.N.; Bouk, S.H.; Ahmed, S.H.; Yaqub, M.A.; Sarkar, M.; Song, H. BEST-MAC: Bitmap-assisted efficient and scalable TDMA-based WSN MAC protocol for smart cities. *IEEE Access* **2016**, *4*, 312–322. [CrossRef]
27. Coman, A.; Sander, J.; Nascimento, M.A. Adaptive processing of historical spatial range queries in peer-to-peer sensor networks. *Distrib. Parallel Databases* **2007**, *22*, 133–163. [CrossRef]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).