Received February 1, 2019, accepted February 21, 2019, date of publication March 4, 2019, date of current version March 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2902621

Lightweight Anonymous Geometric Routing for Internet of Things

YANBIN SUN[1], ZHIHONG TIAN[1], YUHANG WANG[1], MOHAN LI[1], SHEN SU[1], XIANZHI WANG[2], (Member, IEEE), AND DUNQIU FAN[3]

[1]Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou 510006, China
[2]School of Software, University of Technology Sydney, Sydney, NSW 2007, Australia
[3]Threat Intelligence and Network Security Lab, NSFOCUS, Beijing 100089, China

Corresponding authors: Yuhang Wang (wangyuhang@gzhu.edu.cn), Mohan Li (limohan@gzhu.edu.cn), and Shen Su (sushen@gzhu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702223, Grant 61702220, Grant 61871140, Grant 61572153, Grant 61572492, and Grant U1636215, and in part by the National Key Research and Development Plan under Grant 2018YFB0803504 and Grant 2018YEB1004003.

ABSTRACT Mobile service computing relies on efficient and secure data transfer. Geometric routing, which guarantees scalability, efficiency, and mobility, is a promising routing scheme for mobile services in the resource-constrained Internet of Things (IoT). However, the private data transmitted by geometric routing may be eavesdropped by the malicious node and the privacy-preserving becomes one of the important issues for the mobile service in IoTs. Due to the resource constrains, approaches using encryption or hashing are not suitable for geometric routing. In this paper, we propose a lightweight anonymous geometric routing (LAGER) to protect the node-related private data. Instead of using the encryption or hashing approach, LAGER adopts a coordinate confusion mechanism which provides each node an anonymous coordinate of virtual node, such that the private data is decoupled from the node coordinate. The malicious node cannot determine which node the data belongs to even if it obtains the private data. To support the data transmission with anonymous coordinates, LAGER adopts a hybrid routing scheme by combining the greedy routing and a source routing. The experiment results show that LAGER provides effective anonymity with acceptable costs of scalability and efficiency.

INDEX TERMS IoTs, privacy, routing, geometric routing, greedy embedding.

I. INTRODUCTION

During the last decade, Internet of Things (IoT), which interconnects "all" things, has recently abstracted significant attention. Billions of heterogeneous devices including the mobile device in IoT are connected to the internet and collaborate to provide services over wireless networks. The routing of IoT, which supports data transmission, is an important part of mobile service computing. Since most devices in IoT are smart sensor devices or embedded devices, the storage, computing and energy resources of these devices are all constrained, which makes the routing of IoT also faces resource constraint challenges. Fortunately, geometric routing can naturally support the resource constraint of IoT due to its succinctness, efficiency, scalability and mobility.

Geometric routing greedily embeds a topology into a metric space and adopts greedy forwarding for routing. Each node is assigned a virtual coordinate of the metric space and stores the coordinates of its neighbors in its routing table. For greedy forwarding, each node chooses a neighbor, which is the nearest to the destination according to the coordinate distance, as the next hop, and the packet is greedily forwarded to the destination hop-by-hop. The greediness of greedy embedding guarantees that the next hop can always be found.

Since each node only stores its neighbors, and the greedy forwarding takes full use of the shortcuts [1] of the network topology, geometric routing guarantees both efficiency and scalability. Meanwhile, for some geometric routing schemes, such as PIE [2], prefix embedding [3], [4], the structure of coordinate is simple and succinct, which brings lightweight calculations. Geometric routing also supports mobility and it recovers from failures within a very short period of time. All these advantages ensure that geometric routing works well in IoT for mobile services. Geometric routing is not only used for IoT, it is also widely used for F2F (Friend

The associate editor coordinating the review of this manuscript and approving it for publication was Tie Qiu.

to Friend) overlays [5] and ICNs (Information Centric Networks) [6], [7] for content distribution.

Similar to other routing schemes of IoT, geometric routing also faces a security issue, i.e., the privacy disclosure. On one hand, since the node coordinate of geometric routing reveals some topology information, attacker can reconstruct the embedded topology (a spanning tree) based on the coordinates, and then obtains the relationship of any two nodes on the embedded topology by monitoring the packet passes through the attacker. Thus, the location of each node on the embedded topology can be obtained by the attacker. On the other hand, during the node communication, the content of packets may also be illegally obtained by the attacker, which causes the leakage of private data of the two nodes.

To overcome above privacy issues, hashing and encryption are two effective approaches. For the hashing approach, the coordinate can be hashed to an incomprehensible coordinate with a random number [5]. Due to the irreversible property of hash function, it is impossible to derive the node coordinate based on the hashed coordinate. For the encryption approach, the private data can be encrypted during the data transmission [8], such that attackers cannot get the private data even if the encrypted data is captured. However, both hashing and encryption cause serious resource consumption. For each forwarding, each coordinate in a routing table should be hashed. For each packet, the content of packet should be encrypted. In a resource-constrained scenario of IoT, both of the two approaches may not work well.

Not all privacy protection approaches rely on the hashing or encryption. In some application scenarios, the private data is meaningful only when the data is combined with a concrete entity, such as the personal health data on the wearable device. The health data is valuable only when it corresponds to a person who owns the wearable device. Thus, if we decouple this type of private data from its corresponding node, the private data is meaningless, and attackers cannot obtain any valuable information.

In this paper, we propose a lightweight anonymous geometric routing scheme (LAGER) which is suitable for resource-constraint IoT networks. Instead of using encryption and hashing approaches, LAGER adopts a coordinate confusion mechanism such that the real coordinate and data of a node are decoupled. Even if the attacker obtains the entire embedded topology, he still can't know which two nodes are communicating. To support the geometric routing with coordinate confusion mechanism, LAGER adopts a hybrid routing which combines a greedy forwarding with source routing. Theoretical and experimental analyses show that not only the anonymity of geometric routing is guaranteed, but also the scalability and efficiency can be preserved.

The rest of this paper is organized as follows. In Section 2, we discussed the related work of geometric routing. Section 3 reviews geometric routing and presents the detailed design of LAGER with coordinate confusion mechanism and hybrid routing scheme. In Section 4, the performance of LAGER are analyzed and evaluated. Section 5 concludes the paper.

II. RELATED WORK

Mobile services in IoTs are widely used for a variety of smart scenarios, such as the smart city [9], the smart campus [10], the internet of vehicle [11], and so on. To support the mobile service, lots of technologies, such as efficient routing [12], mobile cloud [13], sensor cloud [14], emergency internet of things [15], are adopted for the data transmission and management. Security is also one of the main researches for mobile services in IoTs. For example, the IoT device security [16], network attacks [17], content security [18], and privacy protection, etc. The privacy protection is the main focus of this paper.

The privacy protection focuses on two aspects: the data privacy protection [19], [20] and the location privacy protection [21]. The former is used to hide the private data via the encryption approach when the data is stored in IoT devices or transmitted over the network. The latter is used to prevent the geographic position of service user being leaked. In our paper, we focus on a type of private data which relies on the content and the node location, and it can be viewed as the combination of above two aspects. In addition, the private data is transmitted via the geometric routing and our solution is based on the characteristics of geometric routing.

Early geometric routing schemes [22], [23] adopt physical positions (such as GPS coordinates) as node coordinate for greedy forwarding. These schemes are not suitable for the lightweight anonymous routing. There exist three challenges: (1) Local minimal, a node receives a packet cannot find the next hop which is nearer to the destination than itself. (2) Non-practice, the physical coordinate is difficult to acquire. (3) Inefficiency, the physical coordinate reveals precise location information, the anonymous coordinate should be far away from the real coordinates to ensure anonymity, which causes the cost of efficiency.

To overcome the second challenge, some schemes [24], [25] adopt virtual coordinates for routing. These coordinate is obtained by the embedding technique in some metric spaces. However, there is no theoretical guarantee for greediness of embedding scheme.

To provide theoretical guarantees, Papadimitriou and Ratajczak [26] gave the definition of greedy embedding which supports 100% routing success. Kleinberg [27] proposed a universal greedy embedding approach for arbitrary graphs and achieved a greedy embedding scheme in a Hyperbolic space $\mathbb{H}^2$. He proved that the embedding of a topology can be obtained by the greedy embedding of spanning tree of the topology.

Lots of following researches are based on Kleinberg's work, and most of them focus on the succinctness (measured by the coordinate length), efficiency and scalability of geometric routing. PIE [2] isometrically embedded a spanning tree into $l_\infty^{O(\log n)}$ and the coordinate length is $O(\log^3(n))$ bits in power law graphs with $2 < \lambda < 3$. Hofer et al. [3] first proposed a basic prefix embedding scheme based on PIE, and then optimized the scheme by a virtual tree embedding scheme for content addressing. The coordinate length of

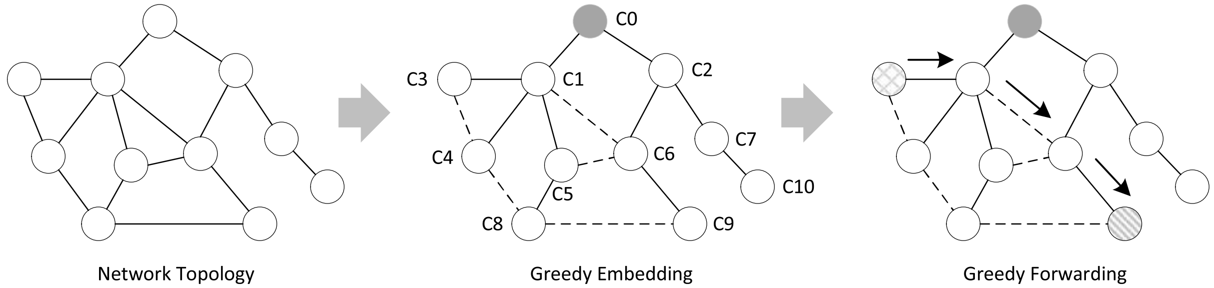


FIGURE 1. **The process of geometric routing.**

optimized scheme is $O(\log^2(n))$ bits in power law graphs with $2 < \lambda < 3$. Only the basic scheme can guarantee the greediness. Our previous work [4] proposes a bit-string prefix embedding scheme, as well as two succinct prefix embedding schemes for geometric routing. All of them can guarantee greediness. The coordinate length of the first scheme is $O(\log^2(n))$ bits in power law graphs with $2 < \lambda < 3$ and the coordinate lengths of the last two schemes are $O(\log^2(n))$ bits for arbitrary graphs. Though all above schemes do not focus on the security of geometric routing, our scheme LAGER is based on above schemes and inherits their good properties.

Only a few researches focus on the security of geometric routing. Some lightweight encryption approaches [28], [29] for IoT are proposed, which can be used for geometric routing. Muthusenthil and Murugavalli [30] proposed a privacy preservation and protection for cluster based geographic routing protocol in MANET using the group signature and encryption. Roos *et al.* [5] proposed anonymous addresses for geometric routing. Each coordinate is set to the same length and each element of coordinate is hashed to an arbitrary bit-string with a random seed. The coordinate distance of greedy forwarding is measured by the hashed coordinates. Different from these security schemes, our scheme LAGER guarantees the anonymity of node via a coordinate confusion mechanism instead of the hashing and encryption approaches.

III. LIGHTWEIGHT ANONYMOUS GEOMETRIC ROUTING

The main purpose of LAGER is to decouple the node-related private data from the corresponding nodes, such that attackers cannot determine which node the private data belongs to. LAGER is based on greedy embedding, but it is not limited to a certain embedding scheme. In this section, we first review the process of greedy embedding, and then present the coordinate confusion mechanism of LAGER, as well as the routing of LAGER.

A. REVIEW OF GREEDY EMBEDDING

1) GEOMETRIC ROUTING

Geometric routing consists of two parts: greedy embedding and greedy forwarding. Fig. 1 shows the process of geometric routing.

The greedy embedding of a topology is generally achieved by greedily embedding the spanning tree of the topology into a metric space. For the greedy embedding in Fig. 1, the solid line constructs a spanning tree, and the dotted line is called *shortcut* (non-tree edge) which is helpful to reduce the routing path length. After the greedy embedding, each node is assigned a coordinate.

Greedy forwarding is based on the embedded topology with tree edges and non-tree edges. As shown in Fig. 1, the packet is greedily forwarded to the destination hop-by-hop. For each time, the neighbor, which is the nearest to the destination, is selected as the next hop. The greedy embedding guarantees that the next hop can always be found.

2) GREEDY EMBEDDING

LAGER is essentially a geometric outing scheme, and it should also embed the topology into a metric space. We first details the definition of greedy embedding, and then review the embedding process. Finally, we present the embedding scheme of LAGER.

The definition of *greedy embedding* is as follows. For a connected graph $G(V, E)$ and a metric space (X, d), the *embedding* of G to X is denoted by a function $f : V \rightarrow X$, where V is the node set of G, E is the edge set of G, and d is the metric (distance function) of X. For $\forall v, u \in G$, $\exists w \in N_v$ (N_v is the neighbor set of v) and $w \neq u$, such that:

$$d(f(v), f(u)) > d(f(w), f(u)),$$

then we call that the embedding f is *greedy*. According to the definition, the greedy embedding can guarantee 100% routing success.

Kleinberg [27] has proved that for a graph G and a spanning subgraph H of G, every greedy embedding of H is also a greedy embedding of G. Thus, the spanning tree T of G is always used to realize the greedy embedding of G.

The greedy embedding of a spanning tree is divided into two steps. First, extract a spanning tree from a topology. There exist multiple distributed protocols for spanning tree extraction in practice, such as [31] and [32]. Second, greedily embed the spanning tree into a metric space. The root node is first assigned an initial coordinate, and then each node computes its coordinate based on the coordinate of its parents, the weight of edge, and other information.

Multiple greedy embedding schemes can be used for LAGER. Here, we adopts our previous work bit-string prefix embedding scheme (Prefix-B) [4]. The metric space of Prefix-B is a bit-string prefix tree. On the prefix tree, each node is assigned a bit-string which is different from that of its sibling. The coordinate of a node is a hierarchical bit-string concatenating bit-strings on the path from the root node to the node.

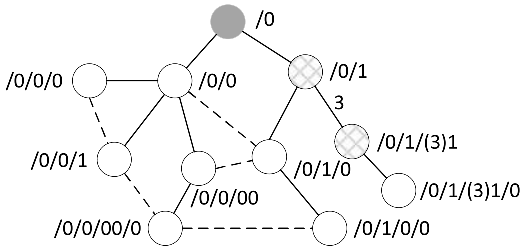


FIGURE 2. Bit-string prefix embedding.

The process of Prefix-B is to assign each node a coordinate of prefix tree. Fig. 2 shows an example of Prefix-B. The root node is first assigned an initial coordinate /0. For each non-root node, it assigned a bit-string by its parent. The coordinate of the node is obtained by appending the bit-string to the coordinate of its parent. If the edge is weighted, the bit-string is also assigned a weight. For example, the node with coordinate /0/1 assigns a bit-string 1 to one of its children. Generally, the coordinate of the child is the combination of /0/1 and 1, i.e., /0/1/1. Since the edge weight between the two nodes is 3, the bit-string should be assigned the weight and the coordinate of the child is /0/1/(3)1.

B. COORDINATE CONFUSION MECHANISM

The coordinate confusion mechanism assigns each node a forged coordinate (we call it an *anonymous coordinate*) which corresponds to a virtual node. The virtual node directly or indirectly connects to a real node, we call the real node an anonymous agent. The anonymous coordinate can be used for routing of LAGER such that the real coordinate node is hidden from the attacker.

Although the anonymous coordinate also contains the topology information, it still guarantees the anonymity of LAGER. For the anonymous coordinate, only a tree structure rather than the entire topology can be inferred. Given an anonymous coordinate, it is difficult for an attacker to find the real coordinate nearby the anonymous coordinate on the topology, because the non-tree edge cannot be revealed by the coordinate and the distance between the real coordinate and the anonymous coordinate may be very large even if the two corresponding nodes are close on topology.

The process of coordinate confusion mechanism is divided into two steps. Each node first determines its anonymous agent, and then obtains its anonymous coordinate from the anonymous agent.

For a node, its anonymous agent is selected by levels which are determined by the number of hops to the node. As shown in Fig. 3, three blue dash lines denote different levels of the grid filled node with hop number 1, 2, 3, respectively. Though a node may belong to multiple levels, our selection strategy can handle this situation.

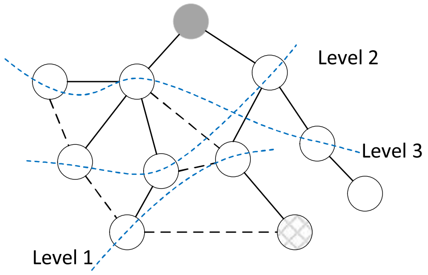


FIGURE 3. Levels of anonymous agent.

Algorithm 1 *Confusion Mechanism (RS)* (at Node u)

```
$p.i$ is the total number of times $p$ should be forwarded;
$p.j$ is the current number of times $p$ has been forwarded;
$p.list$ is a list of passed nodes stored in $p$;
$u.nei$ is the neighbor set of $u$.
if $u.nei \in p.list$ or $p.j = p.i$ then
    $u$ is the confusion node;
else
    $unrd\_nei = u.nei \setminus (p.list \bigcap u.nei)$;
    randomly select a node $v$ from $unrd\_nei$;
    $p.list = p.list \cup \{u\}$;
    $p.j{+}{+}$;
    send the packet $p$ to $v$;
end if
```

Here, we adopt two selection strategies: Random Selection (RS) and Distance-based Selection (DS). For the two strategies, a request packet with a number i is sent to find the anonymous agent. The request contains a number i and other information. The number i denotes on which level the anonymous agent should be selected. The information is used to construct the anonymous coordinate. Meanwhile, the routing path of the request packet should be recorded by each forwarded nodes, such that the anonymous agent can obtain the source routing path to the request node.

For RS, the request node randomly selects a neighbor and sends a request packet to a neighbor. After i times (or there exist no next hop) of forwarding, the anonymous agent can be found. To avoid the request entering a loop, the passed node is recorded in the request packets. If the packet reaches a node with j hops ($j \leq i$) which has no un-recorded neighbor, the forwarding stops and the node is the anonymous agent. Algorithm 1 details the process of RS when a node u receives a request packet p.

For DS, the request node selects a neighbor which is the farthest to the node according to coordinate distance. If some neighbors have the same distance to the node, one of these neighbors is randomly selected. Then the request packet is sent to the neighbor. For each forwarding, the next-hop should also be the farthest to the request node according to coordinate distance. After i times of forwarding, the anonymous agent can be obtained. Algorithm 2 details the process of DS on a node u, where d is coordinate distance and w is the request node.

Algorithm 2 *Confusion Mechanism (DS)* (at Node u)

```
p.i is the total number of times p should be forwarded;
p.j is the current number of times p has been forwarded;
p.list is a list of passed nodes stored in p;
u.nei is the neighbor set of u.
if ($u.nei \in p.list$) or ($p.j = p.i$) or ($\forall v \in u.nei \setminus (p.list \bigcap u.nei)$, $d(v,w) < d(u,w)$) then
    u is the confusion node;
else
    $unrd\_nei = u.nei \setminus (p.list \bigcap u.nei)$;
    select a node $v$ from $unrd\_nei$, such that $d(v,w) \geq d(u,w)$) and $\forall t \in unrd\_nei(t \neq v)$, $d(v,w) > d(t,w)$;
    $p.list = p.list \cup \{u\}$;
    $p.j{+}{+}$;
    send the packet $p$ to $v$;
end if
```

After the anonymous agent of a request node is determined, the anonymous agent produces a virtual node and sends the coordinate of the virtual node (i.e., anonymous coordinate) back to the request node. The anonymous coordinate is produced according to the hop number from the virtual node to the anonymous agent. The hop number is stored in the request packet, such that the request node can adjust the anonymous coordinate for different requirements based on the hop number. The bit-string assigned to virtual node is random produce by the anonymous agent. Fig. 4 shows an example of coordinate confusion of two nodes (the blue node and the red node with solid lines). The anonymous agent is on level 1 with coordinate /0/0/00/0. The coordinate /0/0/00/01/1 of the red node with dashed line is the ananymous coordinate of the red node (solid line) with 2 hops to the anonymous agent. The coordinate distance between the two red nodes is 8 hops, but the distance on the entire topology is only 3 hops. The larger the difference between the two distances is, the better the effect of coordinate confusion mechanism is.

For better anonymous performance, the virtual coordinate can dynamically change on a virtual subtree specified by the anonymous agent, and the real node uses dynamic virtual coordinates within the virtual subtree for routing. Thus, the attacker cannot distinguish which coordinate is virtual and it is more difficult to find the real node.

C. HYBRID ROUTING SCHEME

In LAGER, each node uses its anonymous coordinate for routing, such that the destination of routing is the virtual node from the perspective of the attacker. Since the anonymous coordinate corresponds to a virtual node, the packet can only reach the anonymous agent rather than the real node via greedy forwarding. Thus, LAGER adopts hybrid routing which consists of the greedy routing and the source routing.

To support hybrid routing, each node stores two types of routing information: the coordinates of neighbors and the source routing information. Each node can be an anonymous agent of multiple nodes. The source routing information is the routing path from the anonymous agent to the real node. Such that, when the packet reaches an anonymous agent, a routing path to the real node can be found. Given an anonymous coordinate of a node (the destination), the hybrid routing (the black arrow) is divided into two steps. First, the packet is first greedily forwarded to the anonymous agent hop-by-hop, and then it is transported from the agent to the real node via source routing. As shown in Fig. 5, the block arrow is the process of hybrid routing of LAGER and the blue arrow is the process of greedy routing of general geometric routing. For the hybrid routing, the two steps represented by the dash line and the solid line, respectively.

For any two node v, u, a packet is transported from v to an anonymous coordinate of u. A node w is the anonymous agent of u, then the path length of hybrid routing d_h is as follows,

$$d_h(v,u) = d_g(v,w) + d_s(w,u),$$

where d_g and d_s are the path lengths of greedy routing and source routing, respectively.

Hybrid routing can always find the real destination, i.e., the real node. Since the anonymous agent is the nearest node and the uniquely connected node to the virtual node, all packets to the virtual node is greedily forwarded to the anonymous agent. According to the definition of greedy embedding, the greedy path to the anonymous agent can always be found.

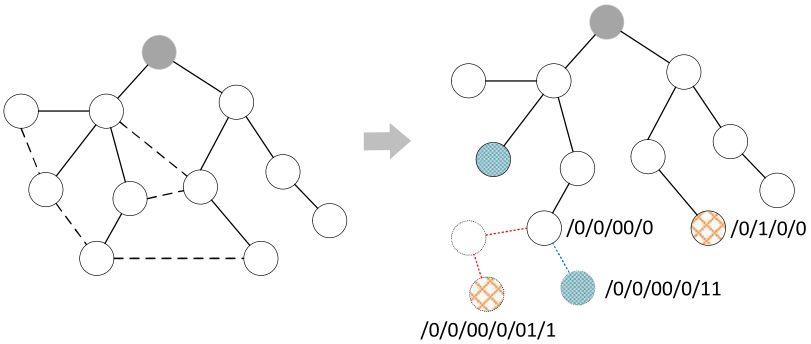


FIGURE 4. An example of coordinate confusion.

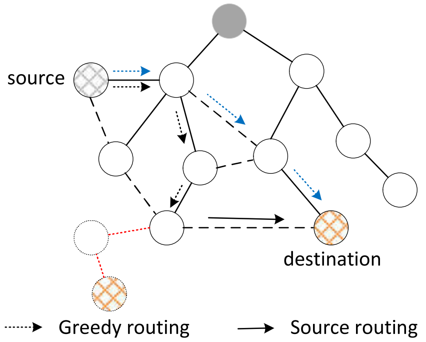


FIGURE 5. General geometric routing and hybrid routing of LAGER.

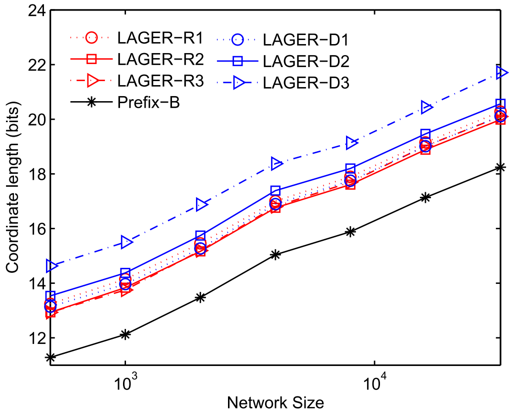


FIGURE 6. The mean coordinate length.

Meanwhile, the agent stores the source routing path to the real node, then the real node can be reached.

IV. EVALUATION AND ANALYSIS

Our schemes are evaluated from three aspects: the scalability, the efficiency and the anonymity. We built a simulator and evaluated LAGER on synthetic topologies. The synthetic topologies are generated by the BA model [33]. We set the initial node number to 3, and set the average node degree to 4, and produced some topologies with different sizes. The power-law of a graph with 5000 nodes is 2.62. Though LAGER supports both the weighted and unweighted graphs, for a better comparison, we set the edge weight to 1.

A. SCALABILITY

According to the hybrid routing, the anonymous coordinate is stored in each packet and the source routing to the real node is in routing tables. Thus, we measured the scalability of LAGER from two aspects: the anonymous coordinate length and the distribution of source routing entries.

The anonymous coordinate length of LAGER is measured with two strategies RS and DS, and it is compared with that of Prefix-B. For each strategy, the level number of anonymous agent is set to 1, 2, 3, respectively. The hop number from the virtual node to the anonymous agent is set to 1.

Fig. 6 shows the mean coordinate length with the growth of topology. Though Prefix-B is the shortest of all schemes, the maximum difference is no more than 4 bits. For LAGER using RS, there is no obvious relationship between the coordinate length and the level. For LAGER with DS, the lower the level is, the shorter the coordinate is. According to the analysis of Prefix-B [4], the coordinate length of Prefix-B is much less than the theoretical result $O(\log^2 n)$, and it has a linear relationship with $O(\log n)$. It is clear that the coordinate length of LAGER also follows the linear relationship.

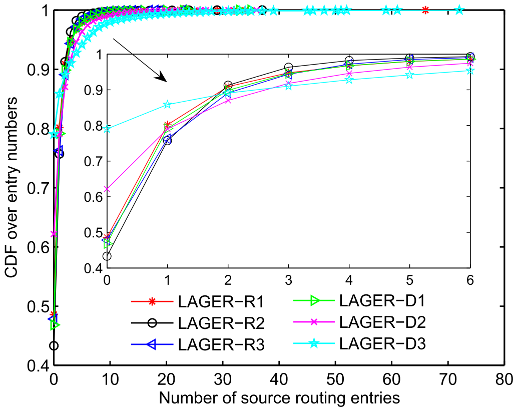


FIGURE 7. Distribution of source routing entries.

Different from general geometric routing, the node of LAGER stores additional routing entries, i.e., the source routing entry. Since each node has one anonymous coordinate, the mean number of source routing entries is 1. Here, we evaluated this aspect via the distribution of source routing entries on a topology with 8000 nodes.

As shown in Fig. 7, the distribution of routing entries is unbalanced. The main reason is that the topology is produced based on the BA model, the distribution of node degree is consistent with the power-law distribution. Thus, the node with high degree has a high probability to be selected as an anonymous agent. For the RS, the distribution of three levels (R1, R2, R3) are close. For the DS, as the level increases, the distribution of routing entries becomes more and more concentrated. In D3, all routing entries are stored in almost 20% nodes. Though the distribution is not uniform, the number of entries each node stores is acceptable. The maximum number is no more than 75 and about 95% of nodes store no more than 6 entries.

B. EFFICIENCY

The efficiency of LAGER is measured by the routing path stretch and the length of routing path.

Path stretch is the ratio of the routing path length to the shortest path length. The lower the path stretch is, the more efficient the routing scheme is. Fig. 8 shows the mean path stretch when the network grows. The path stretch of each scheme remains stable with the growth of topology. Since LAGER cannot directly greedily forward the packet to the real node, the path stretch of Prefix-B is obviously better than that of LAGER. For the LAGER with a strategy RS or DS, the path stretch of low-level is better than that of high-level.

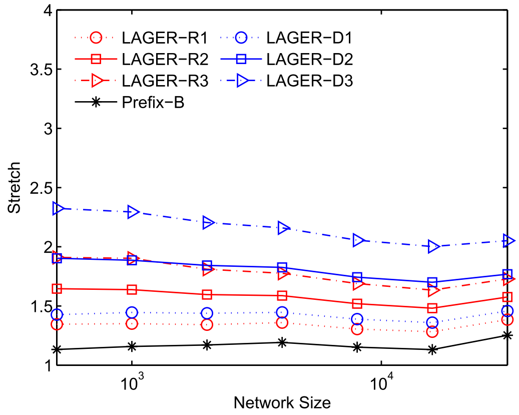


FIGURE 8. Mean path stretch.

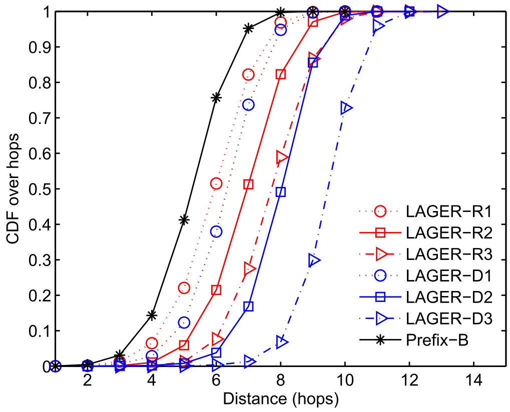


FIGURE 9. Distribution of routing path length.

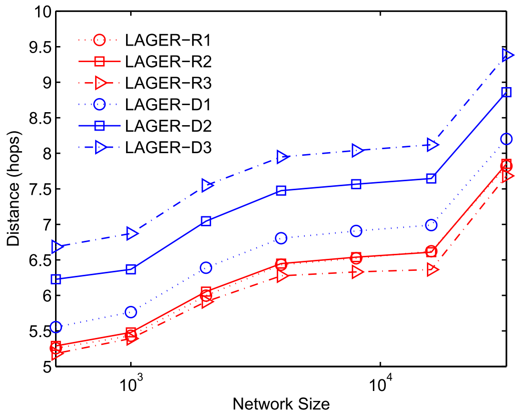


FIGURE 10. Confusion distance.

The routing path of LAGER is measured by hops in a topology with 8000 nodes. Fig. 9 depicts the CDF (Cumulative Distribution Function) of routing hops on the topology. The relationship between any two schemes is the same with that on Fig. 8. Obviously, the routing path of Prefix-B is better than other those of LAGER. The path length difference between Prefix-B and LAGER is little, such as LAGER-R1, LAGER-D1, the path length of LAGER is only 1 hops worse than that of Prefix-B, and the worst case LAGER-D3 is only 4 hops worse than that of Prefix-B. The maximum routing path of LAGER is no more than 13 hops. Overall, the path length of LAGER is adversely affected by the hybrid routing, but routing performance is acceptable.

C. ANONYMITY

The anonymity is measured by the confusion distance which is the coordinate distance from a real node to the virtual node. The larger the confusion distance is, the harder it is for the attacker to find the real node.

Assume that the attacker obtains the whole embedded topology (extended spanning tree with virtual nodes) in the worst case, the anonymity of LAGER can be analyzed from two aspects. First, the virtual node v and the real node u are in the same subtree with u as the root node. Let $sub(u)$ denote the size of subtree, and i denote the distance from v to u. The attacker should find the real node from at least $sub(u)$ nodes. When the children number of node is determined, the $sub(u)$ is positively related to i. Second, u is not the ancestor node of v, i.e., u and v is connected via shortcut edges. Assume that the mean node degree is d, the attacker should find the real node from about $(d^i - 1)/(d - 1)$ nodes, and node number is also positively related to i.

Fig. 10 shows the mean confusion distance with the growth of topology. Obviously, LAGER provides an effective anonymity, the mean confusion distance is more than 5 hops for any cases. The LAGER scheme using DS is much better than that of LAGER using RS. Since the anonymous is randomly selected, the results of schemes of LAGER using RS are about the same. For LAGER using RS, the anonymity (confusion distance) is significant improved when the level increases.

V. CONCLUSIONS

This paper focuses on the privacy-preserving of geometric routing for mobile service computing in resource-constraint IoTs and proposes a lightweight anonymous geometric routing (LAGER) to protect the node-related private data from leaking. Instead of using encryption and hashing approaches, LAGER adopts a coordinate confusion mechanism to decouple the node coordinate and the private data, and adopts a hybrid routing for routing. The attacker obtains the private data without knowing corresponding node has no attack effect. LAGER is not limited to Prefix-B, it is a universal lightweight anonymous approach and can be used for all geometric routing schemes which adopts the spanning tree for greedy embedding. LAGER can guarantee scalability, efficiency and anonymity. The experiment results show that LAGER provides effective anonymity with the cost of scalability and efficiency, the cost is acceptable.

REFERENCES

[1] S. Sahhaf, W. Tavernier, D. Colle, M. Pickavet, and P. Demeester, "Efficient geometric routing in large-scale complex networks with low-cost node design," *IEICE Trans. Commun.*, vol. 99, no. 3, pp. 666–674, 2016.

[2] J. Herzen, C. Westphal, and P. Thiran, "Scalable routing easy as PIE: A practical isometric embedding protocol," in *Proc. 19th IEEE Int. Conf. Netw. Protocols (ICNP)*, Oct. 2011, pp. 49–58.

[3] A. Hofer, S. Roos, and T. Strufe, "Greedy embedding, routing and content addressing for darknets," in *Proc. Conf. Netw. Syst. (NetSys)*, Mar. 2013, pp. 43–50.

[4] Y. Sun, Y. Zhang, B. Fang, and H. Zhang, "Succinct and practical greedy embedding for geometric routing," *Comput. Commun.*, vol. 141, pp. 51–61, Dec. 2017.

[5] S. Roos, M. Beck, and T. Strufe, "Anonymous addresses for efficient and resilient routing in F2F overlays," in *Proc. IEEE 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.

[6] V. Lehman *et al.*, "An experimental investigation of hyperbolic routing with a smart forwarding plane in NDN," in *Proc. IEEE/ACM 24th Int. Symp. Qual. Service (IWQoS)*, Jun. 2016, pp. 1–10.

[7] I. Voitalov, R. Aldecoa, L. Wang, and D. Krioukov, "Geohyperbolic routing and addressing schemes," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 47, no. 3, pp. 11–18, 2017.

[8] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in Internet-of-Things," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1250–1258, Oct. 2017.

[9] J. Qiu, Y. Chai, Y. Liu, Z. Gu, S. Li, and Z. Tian, "Automatic non-taxonomic relation extraction from big data in smart city," *IEEE Access*, vol. 6, pp. 74854–74864, 2018.

[10] Z. Tian *et al.*, "A real-time correlation of host-level events in cyber range service for smart campus," *IEEE Access*, vol. 6, pp. 35355–35364, 2018.

[11] T. Qiu, X. Wang, C. Chen, M. Atiquzzaman, and L. Liu, "TMED: A spider Web-like transmission mechanism for emergency data in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8682–8694, Sep. 2018.

[12] Z. Tian, S. Su, W. Shi, X. Du, M. Guizani, and X. Yu, "A data-driven method for future Internet route decision modeling," *Future Gener. Comput. Syst.*, vol. 95, pp. 212–220, Jun. 2019.

[13] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2015.

[14] M. Li, Y. Sun, Y. Jiang, and Z. Tian, "Answering the min-cost quality-aware query on multi-sources in sensor-cloud systems," *Sensors*, vol. 18, no. 12, p. 4486, 2018.

[15] T. Qiu, R. Qiao, and D. O. Wu, "EABS: An event-aware backpressure scheduling scheme for emergency Internet of Things," *IEEE Trans. Mobile Comput.*, no. 1, pp. 72–84, Jan. 2018.

[16] C. Shi, "A novel ensemble learning algorithm based on DS evidence theory for IoT security," *Comput., Mater. Continua*, vol. 57, no. 3, pp. 635–652, 2018.

[17] Z. Wang, C. Liu, J. Qiu, Z. Tian, X. Cui, and S. Su, "Automatically traceback RDP-based targeted ransomware attacks," *Wireless Commun. Mobile Comput.*, vol. 2018, Dec. 2018, Art. no. 7943586.

[18] T. Qiu, H. Wang, K. Li, H. Ning, A. K. Sangaiah, and B. Chen, "SIGMM: A novel machine learning algorithm for spammer identification in industrial mobile cloud computing," *IEEE Trans. Ind. Informat.*, to be published. doi: 10.1109/TII.2018.2799907.

[19] J. Cui, Y. Zhang, Z. Cai, A. Liu, and Y. Li, "Securing display path for security-sensitive applications on mobile devices," *Comput., Mater. Continua*, vol. 55, no. 1, pp. 17–35, 2018.

[20] M. Hou, R. Wei, T. Wang, Y. Cheng, and B. Qian, "Reliable medical recommendation based on privacy-preserving collaborative filtering," *Comput., Mater. Continua*, vol. 56, no. 1, pp. 137–149, 2018.

[21] Y. Wang, Z. Tian, H. Zhang, S. Su, and W. Shi, "A privacy preserving scheme for nearest neighbor query," *Sensors*, vol. 18, no. 8, p. 2440, 2018.

[22] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, 2000, pp. 243–254.

[23] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wireless Netw.*, vol. 7, no. 6, pp. 609–616, 2001.

[24] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw.*, 2003, pp. 96–108.

[25] A. Caruso, S. Chessa, S. De, and A. Urpi, "GPS free coordinate assignment and routing in wireless sensor networks," in *Proc. 24th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Mar. 2005, pp. 150–160.

[26] C. H. Papadimitriou and D. Ratajczak, "On a conjecture related to geometric routing," in *Algorithms and Experiments for Sensor Systems* (Wireless Networks and Distributed Robotics). Berlin, Germany: Springer, 2004, pp. 9–17.

[27] R. Kleinberg, "Geographic routing using hyperbolic space," in *Proc. 26th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, May 2007, pp. 1902–1909.

[28] S. Singh, P. K. Sharma, S. Y. Moon, and J. H. Park, "Advanced lightweight encryption algorithms for IoT devices: Survey, challenges and solutions," *J. Ambient Intell. Hum. Comput.*, vol. 2017, pp. 1–18, May 2017. doi: 10.1007/s12652-017-0494-4.

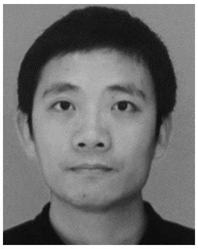
[29] X. Yao, Z. Chen, and Y. Tian, "A lightweight attribute-based encryption scheme for the Internet of Things," *Future Gener. Comput. Syst.*, vol. 49, pp. 104–112, Aug. 2015.

[30] B. Muthusenthil and S. Murugavalli, "Privacy preservation and protection for cluster based geographic routing protocol in MANET," *Wireless Netw.*, vol. 23, no. 1, pp. 79–87, 2017.

[31] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *ACM Trans. Program. Lang. Syst.*, vol. 5, no. 1, pp. 66–77, 1983.

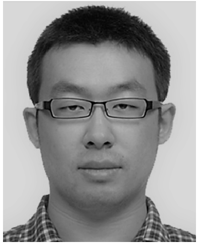
[32] R. Perlman, "An algorithm for distributed computation of a spanningtree in an extended lan," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 15, no. 4, pp. 44–53, 1985.

[33] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.



YANBIN SUN received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China. From 2016 to 2018, he was with Jinan University. He is currently an Associate Professor with Guangzhou University, China. His research interests include network security, future networking, and scalable routing.

ZHIHONG TIAN received the Ph.D. degree. He was a Standing Director of the CyberSecurity Association of China. From 2003 to 2016, he was with the Harbin Institute of Technology. He is currently a Professor, a Ph.D. Supervisor, and a Dean of the Cyberspace Institute of Advanced Technology, Guangzhou University. He was a Member of the China Computer Federation. His current research interests include computer network and network security.



YUHANG WANG received the B.S. degree in computer science from the Heilongjiang University of Science and Technology, Harbin, China, in 2009, and the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2012. He is currently an assistant professor in Guangzhou University, China. His current research interests include network and information security, and location privacy.

MOHAN LI received the B.S., M.S., and Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China. From 2016 to 2018, she was with Jinan University. She is currently an Associate Professor with Guangzhou University, China. Her research interests include data quality and data security.

SHEN SU born in 1985. He received the Ph.D. degree. He is currently an Assistant Professor with Guangzhou University. His current research interest includes inter-domain routing and security.

XIANZHI WANG (M'15) received the bachelor's degree from Xi'an Jiaotong University, Xi'an, China, and the master's and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China. He was a Research Fellow with the Living Analytics Research Centre, Singapore Management University, Singapore, and a Research Associate with the School of Computer Science and Engineering, University of New South Wales, Sydney, and the School of Computer Science, University of Adelaide, Adelaide. He used to visit the Department of Computer Science and Engineering, Arizona State University for one year and an Intern with IBM Research-China for five months. He is currently a Lecturer with the School of Software, University of Technology Sydney, Australia. His research interests include the Internet of Things, artificial intelligence, information fusion, and recommender systems. He was a member of the ACM. He was a recipient of the ARC Discovery Early Career Researcher Award, in 2017 and IBM Ph.D. Fellowship Award, in 2013.

DUNQIU FAN was born in Fujian, China, in 1979. He received the M.E. degree in computer technology with the specialization in computer software development from the China University of Geosciences, Beijing, in 2004. Since 2004, he has been with NSFOCUS Inc. He had been engaged in the Research and Development of intrusion detection system, intrusion prevention system, web application firewall, and then the architecture design of next-generation firewall. Since NSFOCUS founding its threat intelligence team in 2014, he has been the Senior Technical Director, Threat Intelligence and Network Security Lab, NSFOCUS. He has long been engaging in the planning and research in the directions of cutting-edge technologies, such as vulnerability analysis, security event response, intelligent security, and threat intelligence.