sensors

MDPI

Article

A Secure and Privacy-Preserving Navigation Scheme Using Spatial Crowdsourcing in Fog-Based VANETs

Lingling Wang *, Guozhu Liu and Lijun Sun

School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China; LGZ_0228@163.com (G.L.); lijunsun@qust.edu.cn (L.S.)
* Correspondence: teacherwll@163.com; Tel.: +86-136-1642-3910

Academic Editor: Rongxing Lu
Received: 28 December 2016; Accepted: 16 March 2017; Published: 24 March 2017

Abstract: Fog-based VANETs (Vehicular ad hoc networks) is a new paradigm of vehicular ad hoc networks with the advantages of both vehicular cloud and fog computing. Real-time navigation schemes based on fog-based VANETs can promote the scheme performance efficiently. In this paper, we propose a secure and privacy-preserving navigation scheme by using vehicular spatial crowdsourcing based on fog-based VANETs. Fog nodes are used to generate and release the crowdsourcing tasks, and cooperatively find the optimal route according to the real-time traffic information collected by vehicles in their coverage areas. Meanwhile, the vehicle performing the crowdsourcing task can get a reasonable reward. The querying vehicle can retrieve the navigation results from each fog node successively when entering its coverage area, and follow the optimal route to the next fog node until it reaches the desired destination. Our scheme fulfills the security and privacy requirements of authentication, confidentiality and conditional privacy preservation. Some cryptographic primitives, including the Elgamal encryption algorithm, AES, randomized anonymous credentials and group signatures, are adopted to achieve this goal. Finally, we analyze the security and the efficiency of the proposed scheme.

Keywords: fog-based VANETs; real-time navigation; privacy-preserving; spatial crowdsourcing

1. Introduction

Traffic congestion in crowded urban areas has had a number of negative effects on society, such as wasting motorists' time, increasing air pollution from the wasted fuel, and creating a higher chance of collisions, etc. It is reported that commuters in Beijing spent on average 32 min per hour in traffic jams while traveling during rush hours in 2015 [1]. Hence, it is a common experience for a driver to find a better driving route in a congested area. Since real-time traffic information plays a key role in monitoring road conditions and predicting optimal routes of vehicles, it is certainly worth using a real-time navigation system for drivers on the road to find the optimal route of a certain destination.

For the last few years, global positioning system (GPS) [2] technology has been adopted for navigation systems, such as the Autonavi navigation system [3], which provides convenient navigation services based on a local map database. However, since the road conditions are not updated in time in the local map database, conventional GPS-based navigation systems may guide drivers to erroneous routes if some traffic accidents occur in real time.

In the meantime, vehicular ad hoc networks (VANETs), which act as important elements of the intelligent transportation system, has become increasingly popular in many countries. The navigation system based on VANETs can provide more timely and more accurate traffic information. In a typical VANET, vehicles are equipped with on-board units (OBUs) to perform mobile computation and communication with other nearby vehicles, and with road-side units (RSUs) installed along the road. With the support of VANETs and its crowdsensing capability, real-time road conditions can be collected

and transmitted to support the real-time navigation. However, RSUs have limited computation and storage capability, while real-time navigation systems based on crowd sourcing require complex computation and large storage. This challenge has motivated researchers to investigate the new paradigm of VANETs.

Recently, the vehicular cloud has been proposed for big data processing and complicated intelligent analysis on VANET environments [4–6]. However, centralized cloud computing is unnecessary and inefficient for the interactive navigation system. To relieve the computation and communication burden on vehicular cloud, fog computing can be adopted. Fog computing, which was firstly proposed by Cisco in 2012 [7], is an extension of the cloud-based Internet by introducing an intermediate layer between mobile devices and cloud, aiming at the smooth, low-latency service delivery from the cloud to mobile. In this paper, we combine vehicular cloud with fog computing, and consider a new paradigm called fog-based VANETs.

Although fog-based VANETs can provide many possible advantages for real-time navigation systems, several security concerns also have to be addressed before the implementation of the system. A navigation system requires that the identity of the vehicle and communication messages should be authenticated and maintain secrecy to guard against the impersonation, message forgery attacks, and eavesdropping. Meanwhile, privacy preservation must be achieved because the private information, such as vehicles' licenses, location, and speed, etc., needs to be protected. In addition, the authorities should be able to reveal the real identity of the disputed vehicles when necessary.

In this paper, we proposed a secure and privacy-preserving navigation scheme (SPNS) in fog-based VANETs, which use spatial crowdsourcing to collect real-time traffic information and analyze the collected data to provide real-time navigation services to drivers. The security and the privacy preservation of the scheme are also analyzed to evaluate the scheme. In particular, we make the following contributions:

- First, we present a model for a secure navigation scheme in fog-based VANETs, which takes advantage of vehicular cloud and fog computing to make up for the limitation of the previous VANET-based navigation system.
- Second, we construct a specific scheme that can support real-time navigation service to drivers in a congested area. In this way, drivers can quickly find an available route, and, moreover, gasoline and the time wasted in traffic congestion can be reduced. By using the spatial crowdsourcing, the real-time road conditions can be updated in time in fog-based VANETs. Meanwhile, the vehicle performing the crowdsourcing task can get a reasonable reward. Performance analysis shows that the real-time navigation service supported by the proposed scheme is effective.
- Third, the proposed scheme can also ensure the conditional privacy preservation of the vehicles (or drivers), which is regarded as the basic security requirement in VANET communications [8–11].

The rest of the paper is organized as follows: the system model and design goals are described in Section 2. Some preliminaries are given in Section 3. Our scheme is proposed in Section 4. The security analysis is given in Section 5, and the performance analysis is given in Section 6. Related work is reviewed in Section 7. Finally, Section 8 concludes the paper.

2. System Model and Design Goal

In this section, we define the problem by formalizing the system model and the design goal.

2.1. System Model

In this section, we consider the fog-based VANETs and describe our system model, in which communication nodes include trusted authority (TA), navigation servers (NS) and crowdsourcing servers (CS) residing in the fog, and vehicles as shown in Figure 1. The detailed description of system components is as follows:

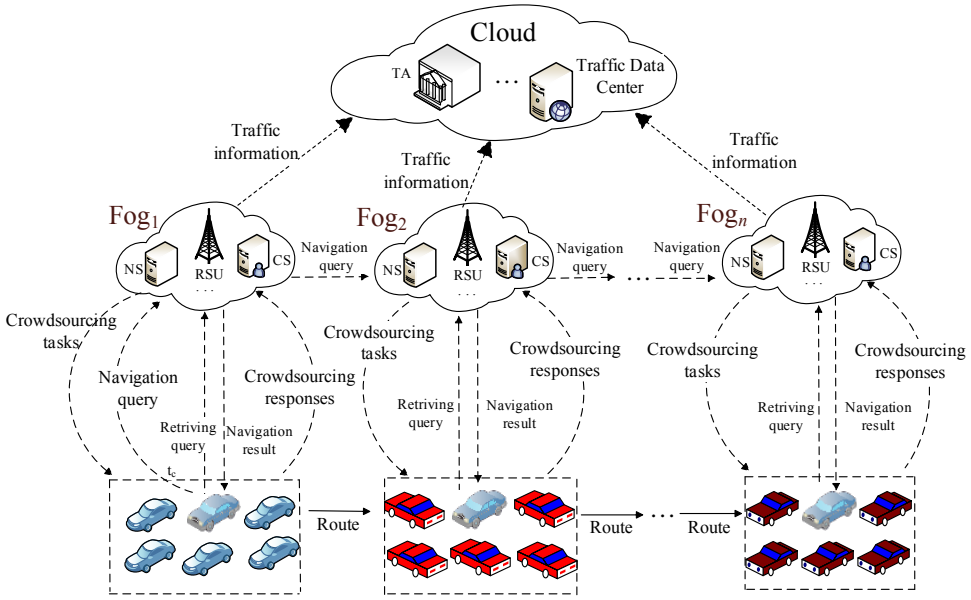

Figure 1. System model.

- TA is trusted and the public agency. For example, the transportation authorities with administrative rights can take on the role of the TA. It is responsible for the registration of fog nodes and vehicles deployed in fog-based VANETs, issuing anonymous credentials and tracing the vehicles' identity in case of rewarding purpose for spatial crowdsourcing, sending fake traffic information for uncongested driving experience, etc.
- The cloud is a set of interconnected computing resources. The cloud provides centralized navigation services for drivers, e.g., Google Map.
- Fog node is a highly virtualized computing system, which is deployed at the edge of networks, e.g., banks, bus terminals, shopping halls, etc. Similar to a lightweight cloud server, fog node is equipped with the on-board large volume data storage, computers and wireless communication facility [12]. In our fog-based VANETs, the fog node consists of navigation servers (NS) and crowdsourcing servers (CS) and conventional RSU, which are in charge of releasing crowdsourcing tasks, computing the optimal path for the querying vehicles, and rewarding the crowdsourcing contributors.
- Vehicles are equipped with irreplaceable and temper-proof OBU device, which enables performing some simple computations, communicating with other vehicles and fog nodes, and these vehicles have a small amount of read-only memory. In our model, OBU is required to generate real-time navigation query, traffic information reports for spatial crowdsourcing tasks, result retrieving query.

As shown in Figure 1, the navigation scheme works as the following. Assume each vehicle and fog node have already registered to the TA. Then, a vehicle can send a navigation query generated by the OBU to the nearby fog node, denoted as fog_1, at time t_0. The navigation server (NS) in Fog_1 forwards the query to the last Fog_n which covers the destination by relaying fog nodes hop by hop. Upon receiving the navigation query, each crowdsourcing server (CS) generates and releases a crowdsourcing task of collecting real-time traffic information to vehicles in its coverage area. In addition, the vehicle who wants to perform the task returns a crowdsourcing report and can get a reasonable reward from the CS. Upon receiving the report, CS verifies it and shares the valuable traffic information with NS. The NS computes the optimal path for the querying vehicle in its area. Meanwhile, NS analyzes and forwards the traffic information to the cloud for other services. Finally, the querying vehicle can get successive navigation results from fog nodes by sending navigation

result retrieving query when entering the coverage of the fog nodes until it ultimately reaches the desired destination.

2.2. Design Goal

Before describing our design goal for the navigation scheme, we first make necessary assumptions in our system model.

Assumption 1. *TA is fully trusted by all vehicles and fog nodes. TA can communicate with fog nodes and vehicles through a secure channel by the internet or any other reliable communication links with high bandwidth. The TA can also inspect all fog nodes and maintain the compromised entities list.*

Assumption 2. *Fog nodes are untrusted. For instance, some honest-but-curious fog nodes may learn the position of specific vehicles and also get some sensitive information for some purposes.*

Assumption 3. *The adversary can overhear V2V (Vehicle to Vehicle) and V2I (Vehicle to Infrastructure) communications to obtain any messages for their purposes, such as tracing the identity of some vehicles. Some dishonest vehicles may overhear the communications to obtain the navigation results queried by other vehicles to enjoy free navigation services if they happen to have the same destination.*

Assumption 4. *TA, fog nodes, and vehicles have clocks for generation of time stamps and check valid time of navigation query and result retrieving query. They can use GPS satellites as a synchronized time source* [13].

Our design goal is to develop a secure and privacy-preserving navigation scheme for vehicles, which can achieve the following desirable requirements: (1) real-time route navigation; (2) authentication; (3) confidentiality; and (4) conditional privacy preservation.

- Real-time path navigation: With the guidance of the fog nodes, a vehicle can conveniently find the optimal path to the desired destination.
- Authentication: Only legitimate entities should take part in the fog-based VANETs. Fog nodes and vehicles should be able to prove themselves by using certificates or credentials. In addition, the origin of the messages should be authenticated to prevent against the impersonation and message forgery attacks. Meanwhile, the identity of the crowdsourcing contributor should be authenticated to get the reward. In addition, only a legitimate subscriber that has service access rights should be able to get navigation service.
- Confidentiality: the navigation query, traffic information report, and navigation result should be kept confidential from eavesdroppers who will illegally use the navigation information for their own purposes.
- Conditional privacy preservation: the real identity of the querying vehicle and the crowdsourcing vehicle should be kept secret. Although the location and destination would be exposed to fog nodes, the adversary can neither link a navigaiton query to a specific vehicle nor identify two navigation queries from the same vehicle. However, once an exceptional event occurs, the fog nodes can learn the vehicles' real identifier with the help of TA.

3. Preliminaries

This section describes some cryptographic primitives which are adopted in our proposed scheme. They are bilinear groups, message-locked encryption, randomized signatures and group signatures.

3.1. Bilinear groups

Bilinear groups are a set of three cyclic groups $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ of prime order q with a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ with the following properties:

1. for all $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_q$, $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$;
2. for $g_1 \neq 1_{\mathbb{G}_1}$ and $g_2 \neq 1_{\mathbb{G}_2}$, $e(g_1, g_2) \neq 1_{\mathbb{G}_T}$;
3. the map e is efficiently computable.

There are three types of pairings defined by Galbraith, Paterson, and Smart [14]: in type 1, $\mathbb{G}_1 = \mathbb{G}_2$; in type 2, $\mathbb{G}_1 \neq \mathbb{G}_2$, but there exists an efficient homomorphism $\phi : \mathbb{G}_2 \to \mathbb{G}_1$ while no efficient one exists in the other direction; in type 3, $\mathbb{G}_1 \neq \mathbb{G}_2$ and there is no efficiently computable homomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$ in either direction. In this paper, we only consider type 3 pairings, which will guarantee the security of the randomized signatures used in our scheme.

3.2. Message-Locked Encryption

A message-locked encryption (MLE) scheme is a symmetric encryption scheme in which the key used for encryption and decryption is itself derived from the message [15]. Instances of this primitive are seeing widespread deployment and application for the purpose of secure deduplication [16–18]. A message-locked encryption scheme MLE = $(\mathcal{P}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{T})$ is a five-tuple of polynomial time algorithm, the last two deterministic:

- On input 1^λ, the parameter generation algorithm $\mathcal{P}$ returns a public parameter P.
- On input P and a message M, the key-generation algorithm $\mathcal{K}$ returns a message-derived key $K \leftarrow \mathcal{K}_P(M)$.
- On input P, K, M, the encryption algorithm $\mathcal{E}$ returns a ciphertext $C \leftarrow \mathcal{E}_P(K, M)$.
- On input P, K and a ciphertext C, the decryption algorithm $\mathcal{D}$ returns $\mathcal{D}_P(K, C) \in \{0,1\}^* \cup \{\perp\}$.
- On input P, C, the tag generation algorithm returns a tag $T \leftarrow \mathcal{T}_P(C)$.

Bellare [15] has summarized four MLE schemes: convergent encryption (CE), Hash-and-CE1 (HCE1), Hash-and-CE2 (HCE2), and randomized convergent encryption (RCE). In our scheme, we utilize the RCE to encrypt the crowdsourcing traffic information report.

3.3. Randomized Signatures

Randomized signature [19] is both an efficient and secure signature with the same features as Camenisch-Lysyanskaya (CL)-signatures [20] but consists of only two elements in the signature. It takes advantage of the full potential of type 3 pairings, in which the space of the signatures and the one of the public key are seperated. A randomized signature scheme usually consists of four algorithms:

- $Setup(1^k)$: given a security parameter k, this algorithm outputs public parameter $param = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e)$. These bilinear groups must be of type 3.
- $Keygen(param)$: selects $g_2 \in \mathbb{G}_2$ and $(x_1, x_2) \in \mathbb{Z}_q^2$, computes $(X_1, X_2) = (g_2^{x_1}, g_2^{x_2})$, and sets sk as (x_1, x_2) and pk as (g_2, X_1, X_2).
- $Sign(sk, m)$: picks a random $g_1 \in \mathbb{G}_1^*$ to compute a signature $\sigma \leftarrow (g, g^{x_1 + m x_2})$.
- $Verify(\sigma, pk, m)$: parses σ as σ_1, σ_2 and checks whether $\sigma_1 \neq 1_{\mathbb{G}_1}$ and $e(\sigma_1, X_1 X_2^m) = e(\sigma_2, g_2)$ are both satisfied. If it is the positive case, it outputs 1, and 0 otherwise.

A randomized signature (σ_1, σ_2) on m can be randomized by selecting a random $r \in \mathbb{Z}_q$ and computing $\sigma' \leftarrow (\sigma_1^r, \sigma_2^r)$, which is still a valid signature on m.

3.4. Group Signatures

The group signature scheme was first introduced by David Chaum and Eugene van Heyst in 1991 [21]. In a group signature scheme, there exists a group manager and several group members, essential to which is a group manager, who is in charge of adding group members and has the ability to reveal the original signer in the event of disputes. A group signature scheme is desired to satisfy three security properties: unforgeability, anonymity, and traceability. Unforgeability ensures that only

the group member can generate signatures on behalf of the group. Anonymity means that signatures do not reveal their signer's identity, except the group manager. Traceability shows that all valid signatures, even those generated by the collusion of multiple group members, can be revoked by the group manager.

4. Proposed Secure and Privacy-Preserving Navigation Scheme

In this section, we present a secure and privacy-preserving navigation scheme (SPNS) in fog-based VANETs, which consists of four parts: (1) system setting; (2) real-time navigation querying; (3) vehicular spatial crowdsourcing; and (4) navigation result retrieving.

4.1. System Setting

Let $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ be three cyclic groups of the same large prime order q. Suppose that $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ are equipped with a type 3 pairing. Let $H : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ be a public collision-resistant hash function. The TA first chooses $(x_1, x_2) \in \mathbb{Z}_q^2$ as the master key and computes $(X_1, X_2, X_3, X_4) = (g_1^{x_1}, g_2^{x_2}, g_1^{x_2}, g_2^{x_1})$ as its public key. In the end, the TA publishes the system parameters $P = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2, X_1, X_2, X_3, X_4)$.

Each fog node has a unique identifier $\mathcal{R}_{ID}$ to identify its position and a map of its coverage. $\mathcal{R}_{ID}$ randomly chooses $y \in \mathbb{Z}_q$ as its secret key and computes $Y = g_2^y$ as its public key. In addition, $\mathcal{R}_{ID}$ maintains a routing table to determine the next fog node to which the vehicle should move forward.

Each vehicle has a unique identifier $\mathcal{V}_{ID}$. The vehicle $\mathcal{V}_{ID}$ randomly chooses $v \in \mathbb{Z}_q$ to compute $V = g_1^v$, $V_1 = X_2^v$, and sends (V, V_1) to the TA to prove its knowledge of v. Then, the TA verifies the validity by checking the equation $e(V, X_2) = e(g_1, V_1)$. If the equation does not hold, the TA returns failure and aborts. Otherwise, the TA picks $s \in \mathbb{Z}_q$ to compute

$$(A, A_1, A_2) \leftarrow (g_1^{\frac{1}{x_1+s}}, g_1^s, (X_1 \cdot V^{x_2})^s).$$

In addition, the TA stores $(\mathcal{V}_{ID}, A)$ in a secure database, and returns (A, A_1, A_2) to $\mathcal{V}_{ID}$ through a secure channel. $\mathcal{R}_{ID}$ sets its secret key as $skv = (v, A, A_1, A_2)$ and the corresponding public key as $pkv = V$.

When $\mathcal{V}_{ID}$ starts to travel in the city, it will generate some short-life keys for navigation queries according to the following steps:

- $\mathcal{V}_{ID}$ chooses m random numbers, $u_1, u_2, \cdots, u_m \in \mathbb{Z}_q^*$ as the short-life private keys and computes the corresponding public keys $U_l = g_2^{u_l}$ for $l = 1, 2, \cdots, m$ for the travel;
- for each short-life public key U_l, $\mathcal{V}_{ID}$ computes the self-delegated certificate $Cert_l$ as follows:
  - randomly choose α, t_α, t_v, $t_\beta \in \mathbb{Z}_q$, compute $T_1, T_2, \beta, \beta_1, \beta_2, \beta_3$ as follows: $T_1 = X_1^\alpha$, $T_2 = A \cdot X_3^\alpha$, $\beta = \alpha \cdot v \bmod q$, $\beta_1 = X_1^{t_\alpha}$, $\beta_2 = T_1^v / X_1^{t_\beta}$, $\beta_3 = e(T_2, g_2^{t_v}) \cdot e(X_3, X_4^{t_\alpha} \cdot g_2^{t_\beta})$;
  - compute $c = H(X_1, X_3, U_l, T_1, T_2, \beta_1, \beta_2, \beta_3)$ and s_α, s_v, s_β where $s_\alpha = t_\alpha + c \cdot \alpha \bmod q$; $s_v = t_v + c \cdot v \bmod q$; $s_\beta = t_\beta + c \cdot \beta \bmod q$; and the certificate of U_l is $Cert_l = \{U_l, T_1, T_2, c, s_\alpha, s_v, s_\beta\}$.
  - anyone can check the validity of $U_l || Cert_l$ by computing: $\beta_1' = X_1^{s_\alpha} / T_1^c$;
    $\beta_2' = T_1^{s_v} / X_1^{s_\beta}$;
    $\beta_3' = e(T_2, g_2^{s_v} \cdot X_4^c) / e(X_3, X_4^{s_\alpha} \cdot g_2^{s_\beta}) e(g_1, g_2^c)$;
    and check whether $c = H(X_1, X_3, U_l, T_1, T_2, \beta_1', \beta_2', \beta_3')$ holds.
- $\mathcal{V}_{ID}$ installs skv and $u_l || U_l || Cert_l$ for $l = 1, 2, \cdots, m$ into the read-only memory of the OBUs.

4.2. Real-Time Navigation Querying

When a vehicle $\mathcal{V}_{ID}^*$ that is equipped with an OBU is driving on the road, it can send a real-time navigation query to the nearby fog node, denoted as $\mathcal{R}_1$. The real-time query utilizes the OBU to generate the navigation information $\{N, U^*, CL, DEST, t_c, t_e\}$, as shown in Table 1.

Using this navigation information, the querier $\mathcal{V}_{ID}^*$ performs the following steps to generate a navigation query:

- utilize Y_1 to encrypt $(U^*, CL, DEST)$ by randomly choosing $k_1 \in \mathbb{Z}_q, g_0 \in \mathbb{G}_1$, and compute $C_1 = g_1^{k_1}, C_2 = g_0 \cdot Y_1^{k_1}, C_3 = AES_{Enc}(g_0, U^*||CL||DEST)$;
- select randomly $(r_1, r_2) \in \mathbb{Z}_q^2$ to compute the randomized signture $(B_1, B_2) \leftarrow (A_1^{r_1}, A_2^{r_1})$, and the hush function $c = H(B_1, B_2, e(B_1, X_2)^{r_2}, N, U^*, CL, DEST, t_c, t_e)$, $\tau = r_2 + c \cdot v$, and output the group signature (B_1, B_2, c, τ);
- finally, send the navigation query Q to the fog node $\mathcal{R}_1$, where $Q = (N, t_c, t_e, C_1, C_2, C_3, B_1, B_2, c, \tau)$.

Upon receiving the navigation query Q, $\mathcal{R}_1$ firstly checks whether the destination $DEST$ is in its coverage. If the answer is yes, it will generate a crowdsourcing task to find the optimal route to the destination for the querying vehicle $\mathcal{V}_{ID}^*$. Otherwise, $\mathcal{R}_1$ performs the following steps to forward Q to the next fog node $\mathcal{R}_2$ according to its routing table. Meanwhile, it will generate a crowdsourcing task to find the optimal route to the next fog node for the querying vehicle.

- Decode $(U^*, CL, DEST) = AES_{Dec}(g_0, C_3)$ by computing $g_0 = C_2 \cdot C_1^{-y_1}$;
- verify the validity of the signature (B_1, B_2, c, τ) by computing

$$B = e(B_1, X_1^c) \cdot e(B_2, g_2^{-c}) \cdot e(B_1, X_2^\tau)$$

  and checking whether the hash $c = H(B_1, B_2, B, N, U^*, CL, DEST, t_c, t_e)$ holds. If not, $\mathcal{R}_1$ returns failure and aborts; Otherwise, it checks the routing table to find the next fog node, denoted as $\mathcal{R}_2$, according to the destination $DEST$.
- $\mathcal{R}_1$ randomly chooses $k_1' \in \mathbb{Z}_q, g_0' \in \mathbb{G}_1$ to compute $C_1' = g_1^{k_1'}$, $C_2' = g_0' \cdot Y_2^{k_1'}$, $C_3' = AES_{Enc}(g_0', U^*||CL||DEST)$;
- finally, $\mathcal{R}_1$ forwards the query $Q' = (N, t_c, t_e, C_1', C_2', C_3', B_1, B_2, c, \tau)$ to $\mathcal{R}_2$.

When $\mathcal{R}_2$ receives Q', it performs the same operations as $\mathcal{R}_1$ and will forward the query to the next fog node until it reaches the last fog node, denoted as $\mathcal{R}_n$, which covers the destination of the querying vehicle $\mathcal{V}_{ID}^*$.

Table 1. The description of navigation information elements.

Element	Description
N	Sequence number: records the query number that is used to distinguish different queries from the same OBU.
U^*	Short-life public key: If a vehicle sends a navigation query at some time, it will randomly choose a short-life public key U^* from the sequence $u_l\|\|U_l\|\|Cert_l$ for $l = 1, 2, \cdots, m$ stored in the OBU. This field is used to record the public key, which will be also used to reward vehicles in the spatial crowdsourcing step.
CL	Current location: records the current position of the querying vehicle on the unique Euclidean plane.
$DEST$	Desired destination: records the destination where the querying vehicle will arrive.
t_c	Current time: records the start querying time.
t_e	Expired time: records the exact time after which the query is invalid, because the life-time of the navigation query is fixed.

4.3. Spatial Crowdsourcing

When receiving a navigation query with a sequence number N, the fog node $\mathcal{R}_j \in \{\mathcal{R}_1, \mathcal{R}_2, \cdots, \mathcal{R}_n\}$ generates and releases a crowdsourcing task of collecting traffic information to all the vehicles in its coverage area. $\mathcal{R}_j$ keeps a tag table TT, which contains all tags of rewarded vehicles for a specific crowdsourcing task. If a vehicle $\mathcal{V}_{ID}^i$ with the secret key $skv_i = (v_i, A_i, A_{1i}, A_{2i})$ wants to perform this task, it performs the operations as follows:

- randomly choose a short-life public key U_i from the $u_l||U_l||Cert_l$ for $l = 1, 2, \cdots, m$ stored in its OBU;
- generate a traffic information report P_i including the current location, current time, driving speed and the road condition;
- randomly choose $(r_{1i}, r_{2i}) \in \mathbb{Z}_q^2$ to calculate the randomized signature $(B_{1i}, B_{2i}) \leftarrow (A_{1i}^{r_{1i}}, A_{2i}^{r_{2i}})$, the hash $c_i = H(N, P_i, B_{1i}, B_{2i}, e(B_{1i}, X_2)^{r_{2i}})$, and $\tau_i = r_{2i} + c_i \cdot v_i$;
- encrypt the traffic information report P_i by using a message-lock encryption algorithm. Choose $L_i \in \mathbb{Z}_q^*$ to compute $K_i = H(P, P_i)$, the tag $T_i = H(P, K_i)$, $E_i = AES_{Enc}(L_i, P_i)$, and $H_i = L_i \oplus K_i \oplus U_i$;
- finally, $\mathcal{V}_{ID}^i$ returns the crowdsourcing response $RP_i = (N, U_i, T_i, E_i, H_i, B_{1i}, B_{2i}, c_i, \tau_i)$ to $\mathcal{R}_j$.

When $\mathcal{R}_j$ receives the crowdsourcing response RP_i, it will compute the optimal path to the destination by the following operations:

- decode the crowdsourcing response P_i by computing $L_i = H_i \oplus K_i \oplus U_i$, $P_i = AES_{Dec}(L_i, E_i)$, and the tag $T_i' = H(P, H(P, P_i))$;
- check whether the equation $T_i' = T_i$ holds. If not, $\mathcal{R}_j$ reject the RP_i. Otherwise, it compares the tag T_i with the element in the tag table TT. If there exists a tag T_j in the tag table TT satisfying $T_i = T_j$, which means the same traffic information report has been stored in the database, $\mathcal{R}_j$ will reject the RP_i. Otherwise, it verifies the signature by computing

$$B_i = e(B_{1i}, X_1^{c_i}) \cdot e(B_{2i}, g_2^{-c_i}) \cdot e(B_{1i}, X_2^{\tau_i})$$

  and checking whether the equation $c_i = H(N, P_i, B_{1i}, B_{2i}, B_i)$ holds. If not, $\mathcal{R}_j$ returns failure and aborts; otherwise, it keeps RP_i in its database;
- $\mathcal{R}_j$ rewards the contributor $\mathcal{V}_{ID}^i$ based on the short-life public key U_i;
- $\mathcal{R}_j$ can compute the optimal path OP_j by using Dijkstra's algorithm in its coverage area;
- choose $k_{1i} \in \mathbb{Z}_q$, $g_{0i} \in \mathbb{G}_2$ to calculate $(e_{1i}, e_{2i}, e_{3i}) = (g_2^{k_{1i}}, g_{0i} \cdot U^{*k_{1i}}, AES_{Enc}(g_{0i}, OP_j))$, and $S_j = U^{*y_j}$. Finally, the navigation result for $\mathcal{V}_{ID}^*$ is $(N, S_j, e_{1i}, e_{2i}, e_{3i})$.

4.4. Navigation Result Retrieval

When the querying vehicle $\mathcal{V}_{ID}^*$ enters the coverage area of $\mathcal{R}_j$, it reads (u^*, U^*) in the OBU, computes $S_j' = Y_j^{u^*}$ and generates the retrieving query RQ.

$\mathcal{V}_{ID}^*$ chooses randomly $(r_1', r_2') \in \mathbb{Z}_q^2$ to compute randomized signature $(B_1', B_2') \leftarrow (A_1^{r_1'}, A_2^{r_1'})$, the hash value $c' = H(t_c, S_j', B_1', B_2', e(B_1', X_2)^{r_2'})$, in which t_c is the current time. $\tau' = r_2' + c' \cdot v'$, and sends $RQ = (t_c, S_j', B_1', B_2', c', \tau')$ to the fog node $\mathcal{R}_j$.

Upon receiving RQ, $\mathcal{R}_j$ computes

$$B' = e(B_1', X_1^{c'}) \cdot e(B_2', g_2^{-c'}) \cdot e(B_1', X_2^{\tau'})$$

and checks whether the equation $c' = H(t_c, S_j', B_1', B_2', B')$ holds. If not, $\mathcal{R}_j$ returns failure and aborts; otherwise, it searches for the navigation result $(N, S_j, e_{1i}, e_{2i}, e_{3i})$ in the database based on S_j'.

$\mathcal{R}_j$ signs the navigation result using its secret key y_j. Randomly choose $r_j \in \mathbb{Z}_q$ to compute $\sigma_1^j = g_1^{r_j}$, $\sigma_2^j = H(N, \mathcal{R}_j, e_{1i}, e_{2i}, e_{3i}, \sigma_1^j)$, $\sigma_3^j = r_j + y_j \cdot \sigma_2^j$. Finally, $\mathcal{R}_j$ sends the navigation result $NR_j = (N, \mathcal{R}_j, e_{1i}, e_{2i}, e_{3i}, \sigma_1^j, \sigma_3^j)$ to $\mathcal{V}_{ID}^*$.

Upon receiving the NR_j, $\mathcal{V}_{ID}^*$ computes $\sigma_4^j = H(N, \mathcal{R}_j, e_{1i}, e_{2i}, e_{3i}, \sigma_1^j)$ and checks whether $\sigma_1^j \cdot U^{*\sigma_4^j} = g_1^{\sigma_3^j}$ holds. If not, $\mathcal{V}_{ID}^*$ returns failure and aborts; otherwise, it decodes $OP_j = AES_{Dec}(g_{0i}, e_{3i})$ by computing $g_{0i} = e_{2i} e_{1i}^{-u^*}$.

4.5. Identity Revocation

Once an accepted message has been disputed, the TA can use the self-delegated certificate $Cert_l = \{U_l, T_1, T_2, c, s_\alpha, s_v, s_\beta\}$ of $(\mathcal{V}_{ID}^*, A)$ to revoke the real identity of the disputed vehicle. The TA uses its secret key $(x_1, x_2) \in \mathbb{Z}_q^2$ to compute

$$T_2^{x_1} / T_1^{x_2} = A^{x_1} \cdot X_3^{x_1 \alpha} / X_1^{x_2 \alpha} = A^{x_1} \cdot g_1^{x_1 x_2 \alpha} / g_1^{x_1 x_2 \alpha} = A^{x_1}$$

and can trace the identity $\mathcal{V}_{ID}^*$ by looking up the entry $(\mathcal{V}_{ID}^*, A)$ in the secure database.

5. Security Analysis

In this section, we discuss security issues of the proposed navigation scheme SPNS, i.e., authentication, confidentiality, and conditional privacy preservation.

(1) Authentication

The identity authentication of vehicles can be guaranteed by the anonymous credentials (A_1, A_2) issued by the TA through the system setting. For the real-time navigation query, vehicles need to generate some anonymous short-life keys U_l by themselves, the authentication of which can also be provided by self-delegated certificates $Cert_l$ created by using authorized key A. Meanwhile, in the spatial crowdsourcing phase, crowdsourcing contributor $\mathcal{V}_i^*$ can get the reward by showing the certificate of U_i used in the crowdsourcing response RP_i. The identity authentication of fog nodes are also guaranteed by the certificates generated by TA.

Message authentication can be guaranteed by using randomized signatures [19] and Schnorr signatures [22]. In the real-time navigation query phase, vehicles generate signature (B_1, B_2, C, τ) of the navigation information by using short randomized signatures [19]. In the spatial crowdsourcing phase, vehicle $\mathcal{V}_i^*$, who wants to perform the crowdsourcing task, returns the crowdsourcing report with a randomized signature $(B_{1i}, B_{2i}, C_i, \tau_i)$. In addition, when a fog node R_j finds the optimal path OP_j, it will generate a signature $S_j = U^{*y_j}$ by using its secret key y_j to sign the OP_j. The security of the signature depends on the discrete algorithm problem in $\mathbb{G}_1$. In the navigation result retrieving phase, vehicles generate signatures of the retrieving query through randomized signatures, and fog nodes create signatures of the navigation result by using Schnorr signatures. Since both short randomized signatures and Schnorr signatures used in our scheme have proven to be unforgeable, the security of signatures generated in our proposed scheme are secure, which guarantees the message authentication.

(2) Confidentiality

To avoid navigation information being illegally obtained by unauthorized vehicles or adversaries, our scheme takes advantage of the Elgamal encryption algorithm, AES algorithm and the message-lock encryption algorithm to encrypt the transmitted information including real-time navigation query, crowdsourcing response and the navigation result. If the encryption algorithms used in the proposed scheme are secure, confidentiality requirements can be satisfied.

First, we consider the anonymous credential. When vehicle $\mathcal{V}_i$ requests an anonymous credential from the TA, it first picks a random number $v \in \mathbb{Z}_q$ to compute $V = g_1^v$, $V_1 = X_2^v$, and sends (V, V_1) to the TA along with a zero-knowledge proof to prove its knowledge of v. Thus, the TA cannot tell the

secret key v of the vehicle, if the discrete algorithm problem in $\mathbb{G}_1$ and $\mathbb{G}_2$ is hard. Then, TA computes the anonymous credential $skv = (v, A, A_1, A_2)$ and sends it to the vehicle through a secure channel, so other vehicles can not illegally receive the anonymous credential by eavesdropping messages from the air.

Second, we consider the navigation query. $\mathcal{V}_i$ utilizes the public key of the fog node Y_1 to encrypt $(U^*, CL, DEST)$ by computing $C_1 = g_1^{k_1}$, $C_2 = g_0 \cdot Y_1^{k_1}$, $C_3 = AES_{Enc}(g_0, U^*||CL||DEST)$ ($k_1 \in \mathbb{Z}_q, g_0 \in \mathbb{G}_1$), which involves the Elgamal encryption algorithm and AES algorithm. Hence, the confidentiality of the navigation query is guaranteed. Similarly, when the fog node can not find the destination in its coverage, it will transmit the navigation query to the next fog node. The fog node will encrypt the $(U^*, CL, DEST)$ by using the Elgamal encryption algorithm and AES algorithm. Therefore, no other vehicle can eavesdrop on the route even if they want to go to the same destination.

Third, we consider the crowdsourcing report. When a vehicle wants to perform the crowdsourcing task, it will generate a traffic information report P_i. To ensure the confidentiality of the report, P_i, we encrypt it by using the message-lock encryption algorithm RCE [15], which can ensure the confidentiality of the report and avoid the repeated rewarding for the same vehicle.

Finally, we consider the navigation result. The navigation result OP_j is encrypted as (e_{1i}, e_{2i}, e_{3i}) by using the public key U_i of the querying vehicle based on Elgamal encryption. When a vehicle asks for the navigation result, it can decode the navigation result by using its short-life secret key u. However, other vehicles can not decrypt the ciphertext $(N, S_j, e_{1i}, e_{2i}, e_{3i})$.

(3) Conditional privacy preservation

In our scheme, although fog nodes can decode $\mathcal{V}_{ID}^*$'s short-life public key, current location, and the destination by computing $(U^*, CL, DEST) = AES_{Dec}(g_0, C_3)$, they can not link this information to some specific vehicle. Because querying vehicle $\mathcal{V}_{ID}^*$ utilizes anonymous credentials (A_1, A_2) to prove itself. To prevent dishonest fog nodes or adversaries from linking the navigation query or the retrieving query to a specific vehicle, $\mathcal{V}_{ID}^*$ provides a randomized version of the credential (A_1, A_2) when generating signatures. Different versions of (A_1, A_2) are unlinkable because linking (A_1, A_2) with (A_1^t, A_2^t) for some $t \in \mathbb{Z}_q$ is equivalent to breaking the DDH assumption in $\mathbb{G}_1$. Furthermore, vehicles use the group signature scheme [21] to sign messages as (B_1, B_2, c, τ), which provides conditional anonymity of the signer.

In addition, $\mathcal{V}_{ID}^*$ takes advantage of group signatures [21] to generate short-life keys $u_l||U_l||Cert_l$ for $l = 1, 2, \cdots, m$ for anonymous authentication in the proposed scheme, so only TA can distinguish the real identity of $\mathcal{V}_{ID}^*$. When vehicle $\mathcal{V}_{ID}^i$ performs the crowdsourcing task, its anonymity and identity privacy can also be guaranteed by the randomly chosen public key U_i.

In conclusion, the anonymity, identity privacy and location privacy of the vehicles have been protected in our scheme. However, once an exceptional event occurs, the fog nodes can learn the vehicle's real identifier with the help of TA. The TA can use the self-delegated certificate $Cert_l = \{U_l, T_1, T_2, c, s_\alpha, s_v, s_\beta\}$ used in the navigation query to trace the identity of the disputed vehicle. Hence, conditional privacy preservation is satisfied in our scheme.

6. Performance Analysis

In this section, we evaluate and compare the computational and communication costs of the proposed scheme SPNS with VSPN (VANET-Based Secure and Privacy-Preserving Navigation Scheme) [23].

Firstly, let T_{PM} denote the time to perform one point scalar multiplication in $\mathbb{G}_1/\mathbb{G}_2$, with T_{AES} the time of AES encryption, T_{par} the time of a pairing operation, respectively. Since these operations dominate the speed of the proposed scheme SPNS, we only consider the time taken by these operations and neglect other operations such as one-way hash function, addition and scalar value manipulation. The number of the operations required in each phase of the proposed SPNS are shown in Table 2.

Next, we consider the computational costs of TA, vehicles and fog nodes in our scheme SPNS compared with VSPN [23]. For the TA side, it is only involved in the system setting and tracing phases. The number of the operations are $8T_{PM} + 2T_{par}$ and $2T_{PM}$ in SPNS, compared to $2T_{PM}$ and $2T_{PM}$ in the protocol VSPN. For the participant TA side, SPNS is less efficient than VSPN because we need more secure parameters for self-delegating short-life public keys and crowdsourcing tasks, which are not mentioned in VSPN. As shown in Table 3, for the vehicles' side, our scheme needs more operations in the system setting phase to generate self-delegating short-life public keys, which can enhance the anonymity of the vehicles and also avoid delegating public key certificates by CA. Since our scheme accomplishes enhanced security and privacy, our scheme needs more operations than VSPN. Considering the experiment in [24] for an MNT (Miyaji, Nakabayashi, Takano) curve [25] with embedding degree $k = 6$, $\mathbb{G}$ being represented by 161 bits and order q being represented by 160 bits, on an Intel Pentium IV 3.0-GHz machine, there exists the following results: $T_{PM} = 0.6$ ms, $T_{par} = 4.5$ ms. Our scheme needs 1.8 ms more to realize the navigation query than VSPN. When retrieving the navigation result from each fog node or RSU, our scheme needs 1.2 ms more than VSPN. For the fog node side, since the computational capability is stronger than common RSU in our fog-based VANET model, it is efficient to realize the operations in each phase of our SPNS scheme.

Table 2. Computational cost of each step in SPNS.

Phases	TA	Fog Node	Vehicle
System setting	$8T_{PM} + 2T_{par}$	T_{PM}	$(2+m)T_{PM} + 2mT_{par}$
Querying	0	$6T_{PM} + 3T_{par} + 2T_{AES}$	$4T_{PM} + T_{AES} + T_{par}$
Crowdsourcing	0	$6T_{PM} + 3T_{par} + 2T_{AES}$	$2T_{PM} + T_{AES}$
Retrieving	0	$4T_{PM} + 3T_{par} + 2T_{AES}$	$n(6T_{PM} + T_{par} + T_{AES})$
Tracing	$2T_{PM}$	0	0

m is the number of the short-life public keys generated by vehicles and n is the number of the fog nodes that relay the navigation query; SPNS is our proposed scheme.

Table 3. Comparison of vehicles' computational cost.

Phases	SPNS	VSPN
Setting	$(2+m)T_{PM} + 2mT_{par}$	$9T_{PM} + T_{par} + 2T_{AES}$
Querying	$4T_{PM} + T_{AES} + T_{par}$	$T_{PM} + T_{AES}$
Crowdsourcing	$2T_{PM} + T_{AES}$	0
Retrieving	$n(6T_{PM} + T_{par} + T_{AES})$	$4nT_{PM}$

m is the number of the short-life public keys generated by vehicles and n is the number of the fog nodes that relay the navigation query; SPNS is our proposed scheme; VSPN is the VANET-Based Secure and Privacy-Preserving Navigation Scheme proposed in [23].

In terms of the communication overhead, VSPN needs the initial RSU, RSU_k, to forward the navigation request Q to its neighbors until Q reaches the last RSU, RSU_d, covering the destination. After RSU_d constructs the navigation reply message, it sends the message back along the reverse path to the initial RSU, RSU_k. Furthermore, the querying vehicle can get the the navigation result from the RSU_k. This procedure needs much communication among RSUs. In contrast to VSPN, our SPNS does not require the fog nodes to return the navigation results to the first fog node. Instead, the querying vehicle can retrieve the navigation result from each fog node and use it to find a proper route to the destination or to the next fog node. In this way, the communication overhead among fog nodes is significantly reduced. Figure 2 shows the comparison results of SPNS and VSPN with respect to the average communication burden between two fog nodes.

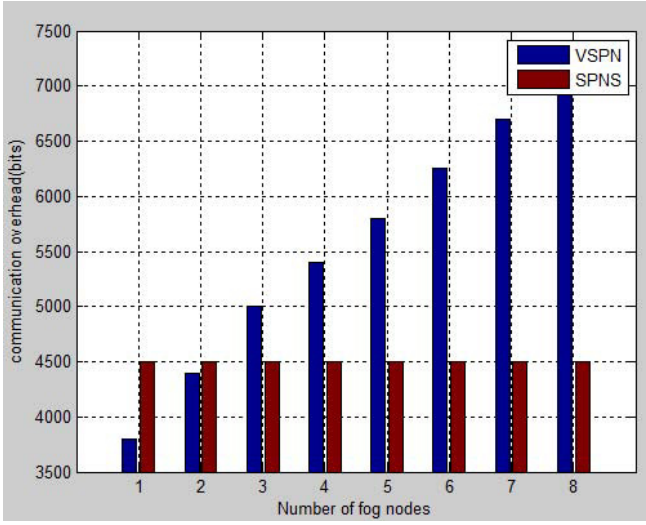

Figure 2. Communication overhead comparison.

7. Related Works

A number of previous studies have been dedicated to designing real-time VANET-based navigation systems for the last few years. In 2009, Lu [26] presented a VANET-based parking navigation protocol, which tracks available parking spaces and guides drivers to the available parking spaces. In their protocol, three RSUs are fully trusted, which provide the navigation functions for a vehicle to find a vacant parking space in a parking lot. However, the protocol [26] can not be used for our navigation purposes. In 2010, another work [27] gave an application of real-time navigation. In addition to driving guidance, the returned routes of the scheme [27] were used for opportunistically routing multimedia information such as images and videos of a desired scene to vehicles. In addition, VANET-based navigation systems [28,29] also emerged to provide real-time navigation services for drivers on roads. By means of the widely deployed vehicular communication infrastructure, the vehicles only needed the OBUs to enjoy the navigation services. However, the security and privacy issues were not concerned in their schemes.

Recently, several VANET-based vehicle navigation systems [23,30–32] have been proposed for drivers' privacy preservation. Chim et al. [23] proposed a VANET-based secure and privacy-preserving navigation system, which utilizes the anonymous credentials to provide secure navigation services to drivers. Based on anonymous credentials and the destination, the system can use the real-time road information to search for an available route for drivers in a distributed way. Nevertheless, this system is vulnerable to insider attacks since the system master key is shared among all vehicles. To eliminate the system master key distribution and simplify the anonymous credential acquisition, Cho et al. [30] introduced a security-enhanced navigation system based on the concept of two person multisignature [33] and identity-based cryptographic schemes [34]. However, how to collect traffic information was not considered in their scheme. Sur et al. [31] pointed out that prior VANET-based secure navigation protocols cannot provide non transferability of anonymous credentials used in their protocols to prevent an insider attacker from sharing her anonymous credentials, and the protocols [23,30] are vulnerable to an attacker who can compromise roadside units (RSUs) deployed on the roads. Sur et al. [31] proposed a secure navigation system based on vehicular cloud from a trapdoor hash function and zero-knowledge proof. However, the anonymous credentials in this system can only be used once for fear of vehicles' sharing credentials with unregistered users. Ni et al. [32] proposed a privacy-preserving real-time navigation system using crowdsourcing. However, their scheme does not take advantage of fog computing, which leads to high efficiency and low-latency.

Our scheme is based on the idea of randomizing anonymous credentials. Once the fog nodes are compromised, they can not link the navigation query or a retrieving query to a specific vehicle.

In this way, it can preserve the privacy of the vehicles. Moreover, the anonymous credentials need not be updated frequently, whereas it can be used for a long time. In our scheme, we utilize fog nodes to issue spatial crowdsourcing tasks to vehicles in their coverage in order to collect real-time road conditions, which guarantee that the retrieving path is real-time and optimal. In addition, the querying vehicle can successively retrieve the navigation result from each fog node when entering its coverage area. This framework is superior to existing solutions, which mainly depend on the assumption that a moving vehicle has to obtain the results from the first fog node, which is quite challenging in reality due to vehicles' high moving speed.

8. Conclusions

In this paper, we proposed a secure and privacy-preserving real-time navigation system based on fog-based VANETs. We utilized the real-time traffic information to guid the vehicle to a desired destination in a distributed way: fog nodes generate the spatial crowdsourcing task to collect real-time road conditions. Then, each fog node takes advantage of the collected traffic information provided by the vehicles in its coverage to compute the optimal route to the destination. Vehicles can get the continuous optimal route from the fog nodes until it arrives at the desired destination. Moreover, the vehicle performing the crowdsourcing task can get a reasonable reward. Our scheme adopts some security primitives to provide a number of security features: (1) vehicles are authenticated by using zero-knowledge proof and randomized anonymous credentials; (2) messages provided by the vehicles and fog nodes can also be authenticated by means of signatures; (3) navigation queries, traffic information report and navigation results are protected from eavesdroppers. Besides satisfying all security requirements, our scheme provides the conditional privacy-preserving requirements. No one including TA can link up a vehicle's navigation query and its identity. However, the TA can trace the identity of the driver who reports false traffic information. Furthermore, our scheme is efficient in terms of computational and communication overhead. For the future work, we will further improve the effectiveness of our scheme and develop a privacy-preserving parking system using vehicular crowdsourcing based on fog-based VANETs.

Acknowledgments: The research was sponsored by the Fund Project of Domestic Visiting Scholars of Excellent Backbone Teachers of Higher Education Institutions in Shandong Province.

Author Contributions: L.L. Wang conceived and designed the schemes; L. Sun performed the experiments; G.Z. Liu analyzed the data; L.L. Wang wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Zheng, J.R. Beijing tops domestic list for traffic congestion. *China Daily*, 27 August 2015.
2. *Global Positioning System Standard Positioning Service Signal Specification*; Navtech GPS Supply: Springfield, VA, USA, 1995.
3. Autonavi Navigation. Availabe online: http://www.autonavi.com/ (accessed on 17 March 2017). (In Chinese)
4. Olariu, S.; Hristov, T.; Yan, G. *The Next Paradigm Shift: From Vehicular Networks to Vehicular Clouds*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2013; pp. 645–700.
5. Yan, G.; Wen, D.; Olariu, S.; Weigle, M.C. Security challenges in vehicular cloud computing. *IEEE Trans. Intell. Transp. Syst.* **2013**, *14*, 284–294.
6. Yu, R.; Zhang, Y.; Gjessing, S.; Xia, W. Toward cloud-based vehicular networks with efficient resource management. *IEEE Netw.* **2013**, *27*, 48–55.
7. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. *Fog Computing and Its Role in the Internet of Things*; Edition of the MCC Workshop on Mobile Cloud Computing; ACM: Helsinki, Finland, 17 August 2012; pp. 13–16.

8. Lin, X.; Sun, X.; Ho, P.H.; Shen, X. GSIS: A secure and privacy-preserving protocol for vehicular communications. *IEEE Trans. Veh. Technol.* **2007**, *56*, 3442–3456.
9. Lin, X.; Sun, X.; Wang, X.; Ho, P.H.; Shen, X. TSVC: Timed efficient and secure vehicular communications with privacy preserving. *IEEE Trans. Wirel. Commun.* **2009**, *7*, 4987–4998.
10. Lin, X.; Lu, R.; Zhang, C.; Ho, P.H.; Shen, X. Security in vehicular ad hoc networks. *IEEE Commun. Mag.* **2008**, *46*, 88–95.
11. Lu, R.; Lin, X.; Zhu, H.; Ho, P.H.; Shen, X. A novel anonymous mutual authentication protocol with provable link-layer location privacy. *IEEE Trans. Veh. Technol.* **2009**, *58*, 1454–1466.
12. Luan, T.H.; Gao, L.; Li, Z.; Xiang, Y.; Wei, G.; Sun, L. Fog computing: Focusing on mobile users at the edge. *arXiv* **2015**, arXiv:1502.01815.
13. Behrendt, K.; Fodero, K. The perfect time: An examination of time-synchronization techniques. In Proceedings of the Distributech, San Diego, CA, USA, 3 January 2005.
14. Galbraith, S.D.; Paterson, K.G.; Smart, N.P. Pairings for cryptographers. *Discr. Appl. Math.* **2008**, *156*, 3113–3121.
15. Bellare, M.; Keelveedhi, S.; Ristenpart, T. Message-Locked Encryption and Secure Deduplication. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, 26–30 May 2013; Volume 7881, pp. 296–312.
16. Hur, J.; Koo, D.; Shin, Y.; Kang, K. Secure Data Deduplication with Dynamic Ownership Management in Cloud Storage. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 3113–3125.
17. Mao, B.; Jiang, H.; Wu, S.; Tian, L. Leveraging Data Deduplication to Improve the Performance of Primary Storage Systems in the Cloud. *IEEE Trans. Comput.* **2013**, *25*, 1775–1788.
18. Yan, Z.; Wang, M.; Li, Y.; Vasilakos, A.V. Encrypted Data Management with Deduplication in Cloud Computing. *IEEE Cloud Comput.* **2016**, *3*, 28–35.
19. Pointcheval, D.; Sanders, O. Short randomizable signatures. In Proceedings of the CT-RSA 2016, San Francisco, CA, USA, 29 February–4 March 2016; Volume 9610, pp. 111–126.
20. Camenisch, J.; Lysyanskaya, A. A signature scheme with efficient protocols. In Proceedings of the International Conference on Security in Communication Networks, Amalfi, Italy, 11–13 September 2002; pp. 268–289.
21. Chaum, D.; Heyst, E.V. Group signatures. In Proceedings of the Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, 8–11 April 1991; Springer: Berlin/Heidelberg, Germany, 1991; pp. 257–265.
22. Schnorr, C.P. Efficient Identification and Signatures for Smart Cards. In *Advances in Cryptology—EUROCRYPT'89*; Springer: Berlin/Heidelberg, Germany, 1990; pp. 688–689.
23. Chim, T.W.; Yiu, S.M.; Hui, L.C.K.; Li, V.O. VSPN: VANET-Based Secure and Privacy-Preserving Navigation. *IEEE Trans. Comput.* **2014**, *63*, 510–524.
24. Scott, M. Efficient Implementation of Cryptographic pairings. Available online: http://ecrypt-ss07.rhul.ac.uk/Slides/Thursday/mscottsamos07.pdf (accessed on 5 June 2016).
25. Miyaji, A.; Nakabayashi, M.; Takano, S. New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2001**, *84*, 1234–1243.
26. Lu, R.; Lin, X.; Zhu, H.; Shen, X. SPARK: A new VANET-based smart parking scheme for large parking lots. In Proceedings of the IEEE INFOCOM, Rio De Janeiro, Brazil, 19–25 April 2009; pp. 1413–1421.
27. Leontiadis, I.; Costa, P.; Mascolo, C. Extending access point connectivity through opportunistic routing in vehicular networks. In Proceedings of the Conference on Information Communications, San Diego, CA, USA, 14–19 March 2010; pp. 1–5.
28. Liu, C.G.; Liu, I.H.; Yang, T.T.; Li, J.S. Navigation-aware association control in vehicular wireless networks. *J. High Speed Netw.* **2013**, *19*, 311–324.
29. Chen, P.Y.; Guo, Y.M.; Chen, W.T. Fuel-saving navigation system in VANETs. In Proceedings of the IEEE Vehicular Technology Conference Fall, Ottawa, ON, Canada, 6–9 September 2010; pp. 1–5.
30. Cho, W.; Park, Y.; Sur, C.; Rhee, K.H. An improved privacy-preserving navigation protocol in VANETs. *J. Wirel. Mob. Netw. Ubiquitous Comput. Dependable Appl.* **2013**, *4*, 80–92.
31. Sur, C.; Park, Y.; Rhee, K.H. An efficient and secure navigation protocol based on vehicular cloud. *Int. J. Comput. Math.* **2016**, *93*, 1–20.

32. Ni, J.B.; Lin, X.D.; Zhang, K.; Shen, X. Privacy-Preserving Real-Time Navigation System Using Vehicular Crowdsourcing. In Proceedings of the IEEE 84th Vehicular Technology Conference (VTC2016-Fall), Montréal, QC, Canada, 18–21 September 2016.
33. Gentry, C.; Silverberg, A. Hierarchical ID-Based cryptography. In *Advances in Cryptology-ASIACRYPT 2002*; Springer: Berlin/Heidelberg, Germany, 2002; pp. 548–566.
34. Chen, L.; Cheng, Z.; Smart, N.P. Identity-based key agreement protocols from pairings. *Int. J. Inf. Secur.* **2007**, *6*, 213–241.