sensors

MDPI

Article

Efficient Privacy-Preserving Data Sharing for Fog-Assisted Vehicular Sensor Networks

Yang Ming * and Xiaopeng Yu

School of Information Engineering, Chang'an University, Xi'an 710064, China; 2017124021@chd.edu.cn
* Correspondence: yangming@chd.edu.cn; Tel.: +86-136-091-16306

Received: 18 December 2019; Accepted: 14 January 2020; Published: 16 January 2020

Abstract: Vehicular sensor networks (VSNs) have emerged as a paradigm for improving traffic safety in urban cities. However, there are still several issues with VSNs. Vehicles equipped with sensing devices usually upload large amounts of data reports to a remote cloud center for processing and analyzing, causing heavy computation and communication costs. Additionally, to choose an optimal route, it is required for vehicles to query the remote cloud center to obtain road conditions of the potential moving route, leading to an increased communication delay and leakage of location privacy. To solve these problems, this paper proposes an efficient privacy-preserving data sharing (EP^{2}DS) scheme for fog-assisted vehicular sensor networks. Specifically, the proposed scheme utilizes fog computing to provide local data sharing with low latency; furthermore, it exploits a super-increasing sequence to format the sensing data of different road segments into one report, thus saving on the resources of communication and computation. In addition, using the modified oblivious transfer technology, the proposed scheme can query the road conditions of the potential moving route without disclosing the query location. Finally, an analysis of security suggests that the proposed scheme can satisfy all the requirements for security and privacy, with the evaluation results indicating that the proposed scheme leads to low costs in computation and communication.

Keywords: vehicular sensor networks; fog computing; data sharing; privacy preserving

1. Introduction

Vehicular sensor networks (VSNs) [1–3], that is, a combination of wireless communication given by vehicular ad hoc networks [4] and the sensing devices installed in the vehicle, can improve traffic conditions in urban cities, and have recently received considerable attention. In VSNs, the vehicles equipped with sensing devices can record a myriad of data reports on the road conditions and environment situations, and these data reports need be uploaded to the remote cloud center [5,6] for processing and analyzing. In addition, vehicles often need to query the road conditions of potential moving routes at remote cloud centers. However, uploading a large amount of data reports to the cloud data center consumes heavy bandwidth, and leads to an increased communication delay.

Recently, fog computing [7] has been proposed to extend the capabilities of cloud computing [8] near vehicles [9], which can locally handle the data reports uploaded by vehicles. These new properties will bring about benefits such as location awareness and low latency. Fog computing has already been used to provide low latency services in vehicular sensor networks, such as navigation services [10] and surface condition monitoring [11].

A typical architecture of fog-assisted vehicular sensor networks (F-VSNs) [12–14] contains the trusted authority, cloud center, fog nodes, and vehicles. The trusted authority is responsible for generating system parameters, and the registration of all entities (cloud center, fog nodes and vehicles). The cloud center provides centralized control with strong computing power and large storage capacity from a remote location. Fog nodes have available computing, storage, and communication resources [15], which is deployed at the edge of networks with physical proximity to vehicles, playing as the bridge across the vehicles and the cloud center. Vehicles are installed with a variety of smart sensors that can sense road conditions and environmental parameters. F-VSNs allows some computations and processing to be performed at the fog nodes, greatly reducing the consumption of communication time and energy.

Although F-VSNs brings a great deal of benefits and conveniences, there still exist several issues in terms of data collection and data query. Specifically, vehicles generate a large amount of sensory data reflecting the road conditions and environment situations, and need to upload the sensory data to cloud center for further processing and analyzing, which brings heavy computation and communication costs. To solve this problem, data aggregation technology, which is designed to aggregate multiple data into one report, has recently received more and more attention.

However, using the existing data aggregation schemes [16–22] cannot determine the number of data reports produced in each road segment, and cannot compute the average sensory data in each road segment. To solve the problem, the scheme [23] exploits the Chinese remainder theorem and Paillier cryptosystem to calculate the average sensory data in each segments; however, it brings heavy computation and communication costs. In addition, to choose an optimal route, vehicles often query about the road conditions of the potential moving routes, but the query reports uploaded by vehicles are tightly associated with the query location, and thus the query location could be disclosed.

The oblivious transfer [24,25], homomorphic encryption technology [26,27], and proxy re-encryption technique [23] have been exploited to hide the query location. However, it is worth noting that the computation and communication costs by the schemes [24,25] is directly proportional to the data dimension, the schemes [26,27] do not support the scenario with high vehicle density, and the scheme [23] needs heavy computation and communication costs.

1.1. Our Contributions

To solve the aforementioned problems, this paper proposes an efficient privacy-preserving data sharing (EP^{2}DS) scheme for fog-assisted vehicular sensor networks. The main contributions of this paper are as follows:

- First, the proposed EP^{2}DS scheme exploits the super-increasing sequence [20] for achieving multi-dimensional data aggregation, while calculating the average sensory data in each road segment, greatly saving on the resources of communication and computation.
- Secondly, by utilizing the modified oblivious transfer [28], the proposed EP^{2}DS scheme is able to query about the road conditions of the potential moving routes without disclosing the query location.
- Thirdly, an analysis of security indicates that the proposed EP^{2}DS scheme is proven to be secure under elliptic curve discrete logarithm (ECDL) assumption in the random oracle model and satisfies all the requirements for security and privacy.
- Finally, the performances of computation and communication in costs are evaluated through quantitative calculations, with the results that the proposed EP^{2}DS scheme is of more efficiency than others.

1.2. Organization

This paper is organized as follows. The related work is surveyed in Section 2. We introduce the background in Section 3. The concrete scheme is proposed in Section 4. Section 5 provides an analysis of the security. In Section 6, the performance evaluation is performed. Section 7 concludes the paper.

2. Related Works

Some works closely related to this paper are briefly reviewed below.

In F-VSNs, massive sensory data is produced in each data dimension, and needs to be uploaded for further processing and analysis; data aggregation schemes [16–23] have received considerable attention recently, and are roughly classified into two categories: single-dimensional data aggregation [16–19] and multi-dimensional data aggregation [20–23]. Zhuo et al. [16] introduced a data aggregation scheme, which protects each involved entity's identity privacy, and allows the requester to examine the correctness of the obtained results. Rabieh et al. [17] employed the data aggregation technique to find out the routes for the vehicle to be in each road segment; however, it only can calculate the data aggregation result, and cannot recover the content in each data dimension.

Xu et al. [18] constructed a privacy-preserving data aggregation scheme that can classify messages based on where and when the sensor data is collected, and aggregate the data collected in the same area and period. Sun et al. [19] designed a data aggregation mechanism considering data integrity and access control. However, the schemes [16–19] are unable to determine the number of the data reports produced in each data dimension, and further fail to calculate the average sensory data in each data dimension. Lin et al. [20] integrated the perturbation technique and super-increasing sequence to combine multiple aggregated data into one data report to improve the energy efficiency.

Lu et al. [21] employed the homomorphic Paillier encryption, one-way hash chain technique and Chinese remainder theorem to achieve lightweight multi-dimensional data aggregation. On the basis of the super-increasing sequence and modified homomorphic Paillier encryption, Wang et al. [22] introduced a multi-subtasks aggregation scheme, in which each aggregated datum is mapped to a specific area and period. Kong et al. [23] designed a privacy-preserving multi-dimensional data sharing scheme using the Chinese remainder theorem and modified Paillier encryption, with counting the number of the sensory data collected at each segments and calculating the average sensory data in each segment.

Although schemes [20–23] are able to calculate the average sensory data in each data dimension, they bring heavy computation costs and communication overhead. In addition, the query vehicle usually wants to know the road conditions of the potential moving route, which could lead to that the query location being disclosed in the data query process, the schemes in [23–27] have been proposed to solve this problem.

Ghinita et al. [24] and Paulet et al. [25] employed the oblivious transfer to hide query location in the data query process, but the communication cost of schemes [24,25] is directly proportional to the data dimension. Zhu et al. [25,26] utilized an improved homomorphic encryption technology to protect the query location in location-based services, but it do not support scenarios with a high vehicle density. Kong et al. [23] utilized the proxy re-encryption technique to hide the query location, but it does not support queries of whole network sensory data during the data query phase.

To sum up, from the review above, the available data aggregation schemes [16–23] either fail to determine the number of data reports produced in each data dimension or bring heavy computation and communication costs. In addition, the communication costs of the existing schemes [23–27] are either directly proportional to the data dimension or bring heavy communication costs in the data query process.

To address the issues above, we propose an EP^{2}DS scheme for fog-assisted vehicular sensor networks, which can not only reduce the computation and communication costs, but also calculate the average sensory data in each road segment. Additionally, the proposed EP^{2}DS scheme can query the road conditions of potential moving routes without disclosing the query location.

3. Background

3.1. System Model

The system model is presented in Figure 1, which is composed of five entities: trusted authority (TA), cloud center (CC), the data collection vehicle V_i $(i = 1, 2, \cdots, \delta)$, fog node FN_j $(j = 1, 2, \cdots, n)$, and the data query vehicle V_q. The road area is divided into m segments, and each segment k $(k = 1, 2, \cdots, m)$ is represented by a unique two-dimensional identifier (u_k, v_k) , approximating of the location coordinates [23]. As to readability, the definitions of notations employed in this study are illustrated in Table 1.

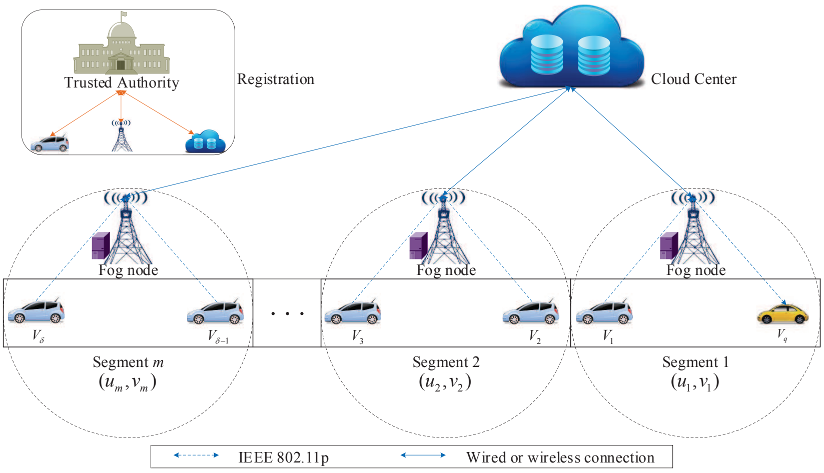


Figure 1. System model.

Table 1. Notations

Symbol	**Definition**
TA	Trusted authority
CC	Cloud center
(s, P_{pub})	TA's public key and private key
(x, P_{cc})	CC's public key and private key
V_i	The i-th data collection vehicle
(ID_i, PID_i)	V_i's real identity and pseudo identity
(x_i, R_i)	V_i's private key
FN_j	The j-th fog node
ID_j	FN_j's identity
(x_{FN_j}, R_{FN_j})	FN_j's private key
V_q	The data query vehicle
(ID_q, PID_q)	V_q's real identity and pseudo identity
(x_q, R_q)	V_q's private key
(u_k, v_k)	Identifier of the segment k
d	Maximum value of sensory data
m	The total number of segments
n	The total number of fog nodes
δ	The total number of vehicles
$\vert d \vert$	Maximum length of sensory data
φ	The vehicles' sharing key
$d^j_{i,k}$	The sensory data captured by V_i at segment k under FN_j
$e^j_{i,k}$	If $d^j_{i,k} > 0$, then $e^j_{i,k}$=1; If $d^j_{i,k} = 0$, then $e^j_{i,k}$= 0.
H_i	Eight one-way hash functions, $H_i : \{0,1\}^* \rightarrow Z_q^*, i = 1,2,\cdots,7$, $H_8 : \{0,1\}^* \rightarrow \{0,1\}^{\vert d \vert -1}$.
$\oplus$	The exclusive OR operation
p, q	Two large prime numbers
F_p	The finite field over p
$\mathbb{G}$	An additive group with the order q on the elliptic curve E over F_p
P	A generator of $\mathbb{G}$

The wireless connections between the vehicles and the fog nodes are brought about by the Institute of Electrical and Electronics Engineers (IEEE) 802.11p standard [29]. The connections between the fog nodes and CC are achieved via either the wired links or other links with low transmission delay and high bandwidth.

TA: A fully trusted entity, which is responsible for the management of the security parameters for the system and the registration of the cloud center, fog nodes, and vehicles, and periodically updates the system information.

CC: An honest-but-curious entity, which is responsible for providing centralized control with powerful storage and computing capabilities from a remote location. In addition, it can perform computational analytics from data reports uploaded by the fog nodes, and distribute data to all fog nodes for further sharing with vehicles [30].

V_i: It is equipped with smart sensors, periodically formatting a data report from the collected sensory data and uploading the data report towards the fog node.

FN_j: This consists of a road side unit and an edge server [13], and aggregates the data reports uploaded by the data collection vehicles under its communication range and transmits the aggregated data report towards *CC*. Meanwhile, each fog node manages one or more segments, and can assist in sharing the sensory data to the query vehicle [31].

V_q: To choose an optimal route, V_q usually sends a query report to the fog node, then the fog node returns a response report to V_q.

In our system model, we assume the fog node is honest-but-curious, i.e., it is able to correctly execute the operations defined in the protocol; however, it also can try to violate the privacy of the vehicle through analyzing the vehicle's data report and query report; meanwhile, we assume neither the fog nodes nor the query vehicles can collude with each other in the proposed EP^{2}DS scheme. Additionally, we assume there exists an attacker, which can eavesdrop on the data transmission and launch attacks.

3.2. Security Requirement

The following security requirements should be achieved.

Authentication and data integrity: The proposed EP^{2}DS scheme should guarantee that any reports are not modified during the transmission process, and can detect any modification of the reports; moreover, any entity in F-VSNs should be able to be authenticated to ensure the reliability of the data source.

Confidentiality: To ensure the privacy of sensory data, the proposed EP^{2}DS scheme should provide confidentiality, i.e., no attacker can obtain the sensory data from data report.

Location privacy preservation: To protect vehicle's query location, it is important not to disclose the query location to fog nodes that provide location-based services in the data query process.

Identity privacy preservation: Apart from the *TA*, any entities should not trace or recognize the identity of the data collection vehicle by analyzing the received data reports.

Traceability: *TA* should be able to reveal the identity of the malicious vehicle uploading the bogus data report.

Unlinkability: Apart from the *TA*, neither fog nodes nor the malicious vehicles can determine whether the two data reports are from the same vehicle.

Resistance to attacks: The proposed EP^{2}DS scheme should be able to withstand various popular attacks such as the modification attack, replay attack, impersonation attack, and man-in-the-middle attack.

3.3. Elliptic Curve

Let F_p be a finite field with a prime number p. The elliptic curve E over F_p defined as the set of all points (x, y) meeting $y^2 = x^3 + ax + b \bmod p$, where $4a^3 + 27b^2 \neq 0$ and $a, b \in F_p$ [32,33].

An infinity point O, and other points on E, form an additive cyclic group $\mathbb{G}$ with the order q and generator P. Let $P \in \mathbb{G}$ and $k \in \mathbb{Z}_q^*$, the scalar multiplication over $\mathbb{G}$ is described as $kP = P + P + \cdots + P$ (k times).

3.4. Security Assumption

ECDL problem [34,35]: Given two elements $P, Q \in \mathbb{G}$, the ECDL problem is to find an integer $x \in \mathbb{Z}_q^*$ such that $Q = xP$.

ECDL assumption [34,35]: It is hard for any probabilistic polynomial-time algorithm to solve ECDL problem with non-negligible probability.

4. The Proposed Scheme

The proposed EP^{2}DS scheme includes system initialization, registration, data collection, and data query phases. Note that the data flows in the data collection and data query phases are shown in Figure 2.

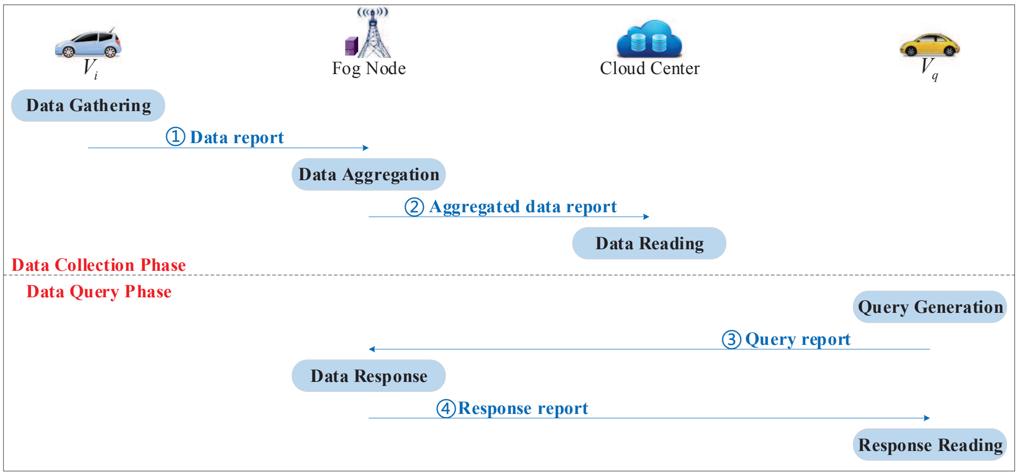


Figure 2. Data flows in the data collection and data query phases.

4.1. System Initialization

TA produces all system parameters through executing the following steps.

(1) *TA* randomly chooses a large prime number p, and selects a non-singular elliptic curve E defined by $y^2 = x^3 + ax + b \bmod p$, where $a, b \in F_p$.
(2) *TA* picks a group $\mathbb{G}$ of E with the prime order q and a generator P.
(3) *TA* randomly chooses $s \in \mathbb{Z}_q^*$ as its master key and computes its public key $P_{pub}{=}sP$.
(4) *TA* chooses eight one-way hash functions $H_i : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$, $i = 1, 2, \cdots, 7$, $H_8 : \{0,1\}^* \rightarrow \in \{0,1\}^{|d|-1}$.
(5) *TA* chooses a super-increasing sequence $\vec{a} = (a_1, a_2, \cdots, a_m)$, such that $\sum_{k=1}^{m} a_k 3n\delta d < q$, $\sum_{k=1}^{i-1} a_k 3n\delta d < a_i$ $(i = 1, 2, \cdots, m)$, where $a_1, a_2, \cdots, a_m$ are large prime numbers and d is the maximum value of the data. Then, *TA* assigns prime number a_k towards segment k.
(6) *TA* publishes the system parameters $\{p, q, \mathbb{G}, P, P_{pub}, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, \vec{a}\}$.

4.2. Registration

All vehicles, fog nodes, and cloud centers register with *TA*.

4.2.1. V_i Registers with TA

(1) V_i sends the identity ID_i to the *TA* in secure channel.
(2) After confirming the identity ID_i, *TA* randomly chooses $w_i \in \mathbb{Z}_q^*$ and computes

$$PID_{i,1} = w_i P,\ PID_{i,2} = ID_i \oplus H_1(w_i P_{pub}, t_i),$$

and sets $PID_i = \{PID_{i,1}, PID_{i,2}, t_i\}$, where t_i represents the valid period of PID_i.

(3) TA randomly chooses $r_i \in \mathbb{Z}_q^*$ and computes

$$R_i = r_i P, x_i = r_i + sH_2(PID_i, R_i, P_{pub}).$$

(4) TA randomly chooses a sharing key $\varphi \in \{0,1\}^{|d|-1}$, and transmits the pseudo identity PID_i, the private key (x_i, R_i) and the sharing key φ to V_i in a secure channel.

4.2.2. FN_j Registers with TA

(1) FN_j sends the identity ID_{FN_j} to the TA in a secure channel.
(2) TA randomly chooses $r_{FN_j} \in \mathbb{Z}_q^*$ and computes

$$R_{FN_j} = r_{FN_j} P, x_{FN_j} = r_{FN_j} + sH_3(ID_{FN_j}, R_{FN_j}, P_{pub}).$$

(3) TA sends the private key (x_{FN_j}, R_{FN_j}) to FN_j in a secure channel.

4.2.3. CC Registers with TA

(1) TA randomly chooses $x \in \mathbb{Z}_q^*$ and computes $P_{cc} = xP$.
(2) TA sends the private key x and public key P_{cc} to CC in a secure channel.

4.3. Data Collection

The data collection phase includes three processes: data gathering, data aggregation, and data reading.

4.3.1. Data Gathering

V_i gathers sensory data in a short period of time, e.g., every five minutes: (i) if there is a sensory data obtained at road segment k under FN_j, i.e., $d_{i,k}^j > 0$, then $e_{i,k}^j$=1; (ii) if there is no sensory data obtained at road segment k under FN_j, i.e., $d_{i,k}^j = 0$, then $e_{i,k}^j$= 0.

V_i produces a data report through executing the following steps:

(1) V_i formats $(d_{i,1}^j, d_{i,2}^j, \cdots, d_{i,m}^j)$ and $(e_{i,1}^j, e_{i,2}^j, \cdots, e_{i,m}^j)$ into $d_i^j = \sum_{k=1}^m a_k(d_{i,k}^j + \varphi)$ and $e_i^j = \sum_{k=1}^m a_k(e_{i,k}^j + \varphi)$.
(2) V_i randomly selects $r_i^j, s_i^j \in \mathbb{Z}_q^*$ and computes

$$A_i^j = r_i^j P, B_i^j = d_i^j P + r_i^j P_{cc}, C_i^j = s_i^j P, D_i^j = e_i^j P + s_i^j P_{cc}.$$

(3) V_i randomly picks $l_i^j \in \mathbb{Z}_q^*$ and calculates

$$L_i^j = l_i^j P, \sigma_i^j = x_i + l_i^j H_4(PID_i, R_i, A_i^j, B_i^j, C_i^j, D_i^j, L_i^j, T_i^j),$$

where T_i^j is current timestamp.
(4) V_i transmits the data report $DR_i^j = \{PID_i, R_i, A_i^j, B_i^j, C_i^j, D_i^j, L_i^j, \sigma_i^j, T_i^j\}$ towards FN_j, as shown in Figure 2 (①).

4.3.2. Data Aggregation

Supposing w vehicles $\{V_1, V_2, \cdots, V_w\}$ upload the data reports $\{DR_1^j, DR_2^j, \cdots, DR_w^j\}$ to FN_j, where $w \leq \delta$. FN_j can aggregate data reports through executing the following steps:

(1) FN_j checks whether t_i is valid and T_i^j is fresh for each $i = 1, 2, \cdots, w$. If t_i is not valid or T_i^j is not fresh, DR_i^j will be rejected. Otherwise, FN_j performs the batch verification using small exponent test [36]. FN_j randomly selects a set of small numbers $\theta_1^j, \theta_2^j, \cdots, \theta_w^j \in [1, 2^w]$ and checks whether the following equation holds

$$\begin{aligned}\textstyle\sum_{i=1}^{w}\theta_i^j\sigma_i^j P = \sum_{i=1}^{w}\theta_i^j R_i + \sum_{i=1}^{w}\theta_i^j H_2(PID_i, R_i, P_{pub})P_{pub}\\ \textstyle+\sum_{i=1}^{w}\theta_i^j H_4(PID_i, R_i, A_i^j, B_i^j, C_i^j, D_i^j, L_i^j, T_i^j)L_i^j.\end{aligned}$$

If it does hold, FN_j computes

$$A^j = \textstyle\sum_{i=1}^{w} A_i^j,\ B^j = \sum_{i=1}^{w} B_i^j,\ C^j = \sum_{i=1}^{w} C_i^j,\ D^j = \sum_{i=1}^{w} D_i^j.$$

(2) FN_j randomly picks $l^j \in \mathbb{Z}_q^*$ and calculates

$$L^j = l^j P, \sigma^j = x_{FN_j} + l^j H_5(ID_{FN_j}, R_{FN_j}, A^j, B^j, C^j, D^j, L^j, T^j),$$

where T^j is current timestamp.

(3) FN_j transmits the aggregated data report $ADR^j = \{ID_{FN_j}, R_{FN_j}, A^j, B^j, C^j, D^j, L^j, \sigma^j, T^j\}$ towards CC, as shown in Figure 2 (②).

4.3.3. Data Reading

After receiving $\{ADR^1, ADR^2, \cdots, ADR^n\}$ from $\{FN_1, FN_2, \cdots, FN_n\}$ respectively, CC executes the following steps:

(1) CC checks whether T^j is fresh for each $j = 1, 2, \cdots, n$. If T^j is not fresh, ADR^j will be rejected. Otherwise, CC randomly chooses a set of small numbers $\theta^1, \theta^2, \cdots, \theta^n \in [1, 2^n]$ and performs the batch verification using small exponent test [36]. CC verifies whether the following equation holds

$$\begin{aligned}\textstyle\sum_{j=1}^{n}\theta^j\sigma^j P = \sum_{j=1}^{n}\theta^j R_{FN_j} + \sum_{j=1}^{n}\theta^j H_3(ID_{FN_j}, R_{FN_j}, P_{pub})P_{pub}\\ \textstyle+\sum_{j=1}^{n}\theta^j H_5(ID_{FN_j}, R_{FN_j}, A^j, B^j, C^j, D^j, L^j, T^j)L^j.\end{aligned}$$

If it does hold, CC calculates

$$\Phi = \textstyle\sum_{j=1}^{n} B^j - x\cdot\sum_{j=1}^{n} A^j,\ \Delta = \sum_{j=1}^{n} D^j - x\cdot\sum_{j=1}^{n} C^j.$$

By solving the discrete log of Φ and Δ with the base P, utilizing the Pollard's lambda algorithm [37], CC can obtain

$$\mu = \textstyle\sum_{j=1}^{n}\sum_{i=1}^{w}(\varphi + d_i^j),\ \nu = \sum_{j=1}^{n}\sum_{i=1}^{w}(\varphi + e_i^j).$$

(2) CC distributes μ and ν to all fog nodes $\{FN_1, FN_2, \cdots, FN_n\}$ for further sharing with vehicles.

4.4. Data Query

The data query vehicle V_q intends to query the data captured at segment c with the identifier (u_c, v_c) at the FN_j. The phase includes three processes: query generation, data response, and response reading.

4.4.1. Query Generation

(1) V_q selects two random numbers $r_q^j, s_q^j \in \mathbb{Z}_q^*$ and calculates

$$E_q^j = r_q^j P,\ F_q^j = u_c P + x_q E_q^j,\ G_q^j = s_q^j P,\ H_q^j = v_c P + x_q G_q^j.$$

(2) V_q randomly picks $l_q^j \in \mathbb{Z}_q^*$ and calculates

$$L_q^j = l_q^j P,\ \sigma_q^j = x_q + l_q^j H_6(PID_q, R_q, E_q^j, F_q^j, G_q^j, H_q^j, L_q^j, T_q^j),$$

where T_q^j is the current timestamp.

(3) V_q transmits the query report $QR_q^j = \{PID_q, R_q, E_q^j, F_q^j, G_q^j, H_q^j, L_q^j, \sigma_q^j, T_q^j\}$ towards FN_j, as shown in Figure 2 (③).

4.4.2. Data Response

(1) After receiving QR_q^j, FN_j checks whether t_q is valid and T_q^j is fresh. If t_q is not valid or T_q^j is not fresh, QR_q^j will be rejected. Otherwise, FN_j verifies whether the following equation holds

$$\sigma_q^j P = R_q + H_2(PID_q, R_q, P_{pub})P_{pub} + H_6(PID_q, R_q, E_q^j, F_q^j, G_q^j, H_q^j, L_q^j, T_q^j)L_q^j.$$

If it does hold, FN_j selects two random numbers $t_q^j, \varphi_q^j \in \mathbb{Z}_q^*$ and calculates

$$J_q^j = t_q^j E_q^j + \varphi_q^j G_q^j, K_q^j = t_q^j F_q^j + \varphi_q^j H_q^j,$$
$$M_q^j = \mu + \sum_{k=1}^m a_k H_8(t_q^j u_k + \varphi_q^j v_k), N_q^j = \nu + \sum_{k=1}^m a_k H_8(t_q^j u_k + \varphi_q^j v_k).$$

(2) FN_j randomly picks $\hat{l}_q^j \in \mathbb{Z}_q^*$ and calculates

$$\hat{L}_q^j = \hat{l}_q^j P, \hat{\sigma}_q^j = x_{FN_j} + \hat{l}_q^j H_7(ID_{FN_j}, R_{FN_j}, J_q^j, K_q^j, M_q^j, N_q^j, \hat{L}_q^j, \hat{T}_q^j),$$

where $\hat{T}_q^j$ is the current timestamp.

(3) FN_j transmits the response report $RR_q^j = \{ID_{FN_j}, R_{FN_j}, J_q^j, K_q^j, M_q^j, N_q^j, \hat{L}_q^j, \hat{\sigma}_q^j, \hat{T}_q^j\}$ towards V_q, as shown in Figure 2 (④).

4.4.3. Response Reading

(1) After receiving RR_q^j, V_q checks whether $\hat{T}_q^j$ is fresh. If $\hat{T}_q^j$ is not fresh, RR_q^j will be rejected. Otherwise, V_q verifies whether the following equation holds

$$\hat{\sigma}_q^j P = R_{FN_j} + H_3(ID_{FN_j}, R_{FN_j}, P_{pub})P_{pub} + H_7(ID_{FN_j}, R_{FN_j}, J_q^j, K_q^j, M_q^j, N_q^j, \hat{L}_q^j, \hat{T}_q^j)\hat{L}_q^j.$$

If it does hold, V_q calculates

$$\Lambda = K_q^j - x_q \cdot J_q^j.$$

By solving the discrete log of Λ with the base P, utilizing the Pollard's lambda algorithm [37], V_q can obtain $\beta_c = H_8(t_q^j u_c + \varphi_q^j v_c)$.

(2) By calling the Algorithm 1, V_q can achieve the average sensing data $\overline{d}_c$ captured at segment c.

Algorithm 1 Recovery $\overline{d}_c$ captured at segment c

```
Input: $(a_1, a_2, \cdots, a_m)$, $\beta_c$, $\varphi$, $\delta$, $M_q^j$ and $N_q^j$
Output: $\overline{d}_c$
begin:
    set $X_1 = M_q^j, X_2 = N_q^j;$
    for $k = m$ to $c$ do
        $d_k = \frac{X_1 - X_1 \bmod a_k}{a_k}, e_k = \frac{X_2 - X_2 \bmod a_k}{a_k};$
        $X_1 = X_1 \bmod a_k, X_2 = X_2 \bmod a_k;$
    return $\overline{d}_c = \frac{d_c - \beta_c - \delta\varphi}{e_c - \beta_c - \delta\varphi}$.
end
```

5. Security

This section depicts the security proof of the proposed EP^{2}DS scheme in the random oracle model. Additionally, a security evaluation and comparison on the proposed EP^{2}DS scheme and schemes of [17,19,23,25,26] is conducted.

5.1. Security Model

The security model of the proposed EP^{2}DS scheme can be found in the Appendix A.

5.2. Security Proof

The security proof of the proposed EP^{2}DS scheme can be found in the Appendix B.

5.3. Analysis and Comparison of Security Requirement

Authentication and data integrity: Based on Theorem 2, no polynomial-time attacker is able to fake a valid data report owing to the ECDL assumption. Therefore, authentication and data integrity can be ensured in the proposed EP^{2}DS scheme.

Confidentiality: Based on Theorem 1, without the cloud center's private key x, any attacker is unable to compute the sensing data $\mu = \sum_{j=1}^{n} \sum_{i=1}^{w} (\varphi + d_i^j)$ and $\nu = \sum_{j=1}^{n} \sum_{i=1}^{w} (\varphi + e_i^j)$, and thus confidentiality can be ensured in the proposed EP^{2}DS scheme.

Location privacy preservation: Based on Theorem 1, without the the data query vehicle's private key x_q, no attacker can obtain the query location (u_c, v_c) from $\{E_q^j = r_q^j P, F_q^j = u_c P + x_q E_q^j, G_q^j = s_q^j P, H_q^j = v_c P + x_q G_q^j\}$, and hence the location privacy can be guaranteed in the proposed EP^{2}DS scheme.

Identity privacy preservation: On the basis of the proposed EP^{2}DS scheme, the identity ID_i of V_i is only contained in the pseudo identity $PID_i = \{PID_{i,1}, PID_{i,2}, t_i\}$, where $PID_{i,1} = w_i P$, $PID_{i,2} = ID_i \oplus H(w_i P_{pub}, t_i)$ and $P_{pub} = sP$. To extract the identity ID_i of V_i, the attacker has to compute $ID_i = PID_{i,2} \oplus H(s \cdot PID_{i,2}, t_i)$. However, it is impossible to solve $w_i \cdot s \cdot P$ for any attacker to obtain ID_i without knowing w_i and s. Therefore, the identity privacy is guaranteed in the proposed EP^{2}DS scheme.

Traceability: In accordance with the proposed EP^{2}DS scheme, TA can adopt its own master key s to calculate $ID_i = PID_{i,2} \oplus H(s \cdot PID_{i,2}, t_i)$, and find out the identity ID_i of V_i from the pseudo identity PID_i involved in the data report, with the proposed EP^{2}DS scheme satisfying the traceability.

Unlinkability: On the basis of the proposed EP^{2}DS scheme, the data reports generated by any vehicle are random, and any attacker cannot link the two data reports sent by the same vehicle, with the proposed EP^{2}DS scheme realizing the traceability.

Resistance to attacks: The proposed EP^{2}DS scheme is able to withstand the networks attacks in the following:

- **Modification attack:** Based on Theorem 2, any polynomial attacker is unable to forge a valid data report with modification on data reports found.
- **Replay attack:** On the basis of the proposed EP^{2}DS scheme, the timestamp is contained in the data report. By examining freshness of the timestamp, the verifier is able to bear any replay attacks.
- **Impersonation attack:** From Theorem 2, no attacker can fabricate a legal data report without vehicle's private key.
- **Man-in-the-middle attack:** The analysis of the modification attack shows that any modification of the data reports on transmission is able to be found.

Security comparisons of schemes [17,19,23,25,26] and the proposed EP^{2}DS scheme are displayed in Table 2. S1, S2, S3, S4, S5, S6, S7, S8, S9, and S10 are used to represent authentication and data integrity, confidentiality, location privacy preservation, identity privacy preservation, traceability, unlinkability, the modification attack, the replay attack, the impersonation attack, and the man-in-the-middle attack, respectively.

Table 2. Security comparisons. Efficient privacy-preserving data sharing (EP^{2}DS), $\surd$ represents "satisfy" and $\times$ denotes "does not satisfy".

Security	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**
Rabieh et al.'s scheme [17]	$\surd$	$\surd$	$\times$	$\times$	$\times$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
Sun et al.'s scheme [19]	$\surd$	$\surd$	$\times$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
Kong et al.'s scheme [23]	$\surd$	$\surd$	$\surd$	$\times$	$\times$	$\surd$	$\surd$	$\times$	$\surd$	$\times$
Paulet et al.'s scheme [25]	$\times$	$\surd$	$\surd$	$\times$	$\times$	$\surd$	$\times$	$\times$	$\times$	$\times$
Zhu et al.'s scheme [26]	$\surd$	$\surd$	$\surd$	$\times$	$\times$	$\surd$	$\surd$	$\times$	$\surd$	$\times$
EP^{2}DS	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$

In accordance with Table 2, Rabieh et al.'s scheme [17] is able to provide location privacy preservation, identity privacy preservation, and traceability. Sun et al.'s scheme [19] cannot achieve location privacy preservation. Kong et al.'s scheme [23] cannot achieve identity privacy preservation, traceability, the replay attack, and the man-in-the-middle attack. Paulet et al.'s scheme [25] cannot achieve authentication and data integrity, identity privacy preservation, traceability, the modification attack, the replay attack, the impersonation attack, and the man-in-the-middle attack. Zhu et al.'s scheme [26] cannot achieve identity privacy preservation and traceability, the replay attack, and the man-in-the-middle attack. In contrast, all security requirements are able to be satisfied in the proposed EP^{2}DS scheme.

6. Performance Evaluation

We analyze the computation and communication costs of these schemes [17,19,23,25,26] and the proposed EP^{2}DS scheme, and evaluate their performance.

To realize a fair comparison, we compare these schemes [17,19,23,25,26] with the proposed EP^{2}DS scheme under the 80-bit security level [38]. Regarding the pairing-based schemes [17,19,23,25,26], we choose a bilinear pairing $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, where $\mathbb{G}_1$ is an additive group defined by the generator P with order q on the super singular elliptic curve $E : y^2 = x^3 + x \bmod p$ with the embedding degree 2, q is 160-bit Solinas prime number and p is 512-bit primer number meeting $q \cdot 12 \cdot r = p + 1$. With regard to the proposed EP^{2}DS scheme, we pick a group $\mathbb{G}$, where $\mathbb{G}$ is produced by the generator P with the order q on an elliptic curve $E : y^2 = x^3 + ax + b \bmod p$ with a prime order q, where q, p are 160 bits prime number and $a = -3$, b is 160-bits random prime number.

The running time of the operations is able to be derived by making use of the MIRACL Crypto SDK [39]. We run the experiment on a 64-bit Windows 10 operating system with 2.53 GHz, an i7 CPU and 4 GB memory. Table 3 lists the average running time for these operations.

Table 3. Runtime of cryptographic operation (millisecond).

Notations	**Descriptions**	**Runtime**
T_{sm}	Scalar multiplication operation in $\mathbb{G}$	0.3851
$T_{\log}$	Solving the DL operation mod p	0.6438
T_e	The exponentiation operation in $\mathbb{G}_1$	2.0289
T_m	The multiplication operation in $\mathbb{G}_1$	1.4293
T_h	Map to point hash function operation	3.5819
T_p	Bilinear pairing operation in $\mathbb{G}_1$	10.3092

6.1. Computation Costs

The computation costs of the proposed EP^{2}DS scheme and these schemes [17,19,23,25,26] are displayed in Table 4.

Table 4. Comparison of computation costs.

Scheme	Data Collection Phase			Data Query Phase	
	V_i	FN	CC	V_a	FN
[17]	$2T_m$+$2T_e$ = 6.9164 ms	T_m+T_e+$(w$+$1)T_p$ = 10.3092w+13.7674 ms	T_e+$(n$+$1)T_p$ =10.3092n+2.0289 ms	–	–
[19]	$2T_m$+T_e+T_h = 15.1967 ms	$(w$+$3)T_m$+$4T_p$ = 1.4293w+45.5247 ms	T_m+nT_e+$2T_p$ =2.0289n+11.7385 ms	–	–
[23]	$4T_m$+$4T_e$ = 13.8328 ms	$2wT_m$ = 2.8586w ms	$6nT_m$+$4nT_e$ =16.6914n ms	$10T_m$+$7T_e$ =28.4953 ms	$9T_m$+$7T_e$ =27.0660 ms
[25]	–	–	–	$5T_m$+$9T_e$ =25.4066 ms	6mT_m+(8m+3)T_e =24.8070m+6.0867 ms
[26]	–	–	–	$2T_p$+$5T_e$ =30.7629 ms	$4T_p$+$4T_m$ =46.9540 ms
EP^{2}DS	$5T_{sm}$ =1.9255 ms	$(w$+$3)T_{sm}$ =0.3851w+1.1553 ms	$(n$+$3)T_{sm}$+$2T_{\log}$ =0.3851n+2.4429 ms	$11T_{sm}$+$2T_{\log}$ =5.5237 ms	$8T_{sm}$ =3.0808 ms

In the data collection phase, for Rabieh et al.'s scheme [17], V_i requires running two multiplication operations in $\mathbb{G}_1$ and two exponentiation operations in $\mathbb{G}_1$, thus the total time is $2T_m + 2T_e$ = 6.9164 ms. FN requires executing one multiplication operation in $\mathbb{G}_1$, one exponentiation operation in $\mathbb{G}_1$, and $w+1$ bilinear pairing operations in $\mathbb{G}_1$, and thus the total time is $T_m + T_e + (w+1)T_p$ = 10.3092w+13.7674 ms. CC requires executing one exponentiation operation in $\mathbb{G}_1$ and $n+1$ bilinear pairing operations in $\mathbb{G}_1$, and hence the total time is $T_e + (n+1)T_p = 10.3092n + 2.0289$ ms.

For Sun et al.'s scheme [19], V_i requires running two multiplication operations in $\mathbb{G}_1$ and one exponentiation operation in $\mathbb{G}_1$ and one map to point hash function operation, thus the total time is $2T_m + T_e + T_h$ = 15.1967 ms. FN requires executing $w+3$ multiplication operations in $\mathbb{G}_1$ and four bilinear pairing operations in $\mathbb{G}_1$, so the total time is $(w+3)T_m + 4T_p$ = 1.4293w +45.5247 ms. CC requires executing one multiplication operation in $\mathbb{G}_1$, n exponentiation operations in $\mathbb{G}_1$ and two multiplication operations in $\mathbb{G}_1$, and hence the total time is $T_m + nT_e + 2T_p = 2.0289n + 11.7385$ ms.

For Kong et al.'s scheme [23], V_i requires running four multiplication operations in $\mathbb{Z}_{n^2}$ and four exponentiation operations in $\mathbb{Z}_{n^2}$, thus the total time is $4T_m + 4T_e$ = 13.8328 ms. FN requires executing $2w$ multiplication operations in $\mathbb{G}_1$, so the total time is $2wT_m$ = 2.8586w ms. CC requires executing $6n$ multiplication operations in $\mathbb{G}_1$ and $4n$ exponentiation operations in $\mathbb{G}_1$, and hence the total time is $6nT_m + 4nT_e = 16.6914n$ ms.

For the proposed EP^{2}DS scheme, V_i needs to run five scalar multiplication operations in $\mathbb{G}$, and therefore the total time is $5T_{sm}$ = 1.9255 ms. FN requires executing $w+3$ scalar multiplication operations in $\mathbb{G}$; accordingly, the total time is $(w+3)T_{sm}$ = 0.3851w+1.1553 ms. CC requires executing $n+3$ scalar multiplication operations in $\mathbb{G}$ and two solving the DL operations; therefore, the total time is $(n+3)T_{sm} + 2T_{log}$ = 0.3851n+2.4429 ms.

In the data query phase, for Kong et al.'s scheme [23], V_q requires running ten multiplication operations in $\mathbb{G}_1$ and seven exponentiation operations in $\mathbb{G}_1$, so the total time is $10T_m + 7T_e$ = 28.4953 ms. FN needs to run nine multiplication operations in $\mathbb{G}_1$ and seven exponentiation operations in $\mathbb{G}_1$, the total time is thus $9T_m + 7T_e$ = 27.0660 ms. For Paulet et al.'s scheme [25], V_q requires running five multiplication operations in $\mathbb{G}_1$ and nine exponentiation operations in $\mathbb{G}_1$, the total time is thus $5T_m + 9T_e$ = 25.4066 ms. FN needs to run $6m$ multiplication operations in $\mathbb{G}_1$ and $8m+3$ exponentiation operations in $\mathbb{G}_1$, the total time is thus $6mT_m + (8m+3)T_e$ = 24.8070m +6.0867 ms.

For Zhu et al.'s scheme [26], V_q requires running five exponentiation operations in $\mathbb{G}_1$ and two bilinear pairing operation in $\mathbb{G}_1$, the total time is thus $5T_e + 2T_p$ = 30.7629 ms. FN needs to run four multiplication operations in $\mathbb{G}_1$ and four bilinear pairing operation in $\mathbb{G}_1$, the total time is thus $4T_m + 4T_p$ = 46.9540 ms.

For the proposed EP^{2}DS scheme, V_q needs to run eleven scalar multiplication operations in $\mathbb{G}$ and two solving the DL operations, and hence the total time is $11T_{sm} + 2T_{log}$ = 5.5237 ms. FN needs to run eight scalar multiplication operations in $\mathbb{G}$, thus the total time is $8T_{sm}$ = 3.0808 ms.

Figure 3 clearly demonstrates the comparison result of computation costs in the data collection phase. Figure 3a shows that the computation costs of V_i is 1.9255 ms, which decreases by 72.2%, 87.3%, and 86.1% compared with that by Rabieh et al.'s scheme [17], Sun et al.'s scheme [19], and Kong et al.'s scheme [23], respectively. As shown in Figure 3b, the computation costs of FN increase linearly with the number of vehicles, with the proposed EP^{2}DS scheme having a lower slope compared with Rabieh et al.'s scheme [17], Sun et al.'s scheme [19], and Kong et al.'s scheme [23]. From Figure 3c, we can see that the computation costs of CC grows linearly with the number of fog nodes, and the proposed EP^{2}DS scheme has a lower slope compared with Rabieh et al.'s scheme [17], Sun et al.'s scheme [19], and Kong et al.'s scheme [23].

Figure 4 clearly indicates the comparison result of the computation costs in the data query phase. From Figure 4a, we can know that the computation costs of V_q in the proposed EP^{2}DS scheme are 5.5237 ms, which decreases by 80.6%, 78.3%, and 82.0% compared with that by Kong et al.'s scheme [23], Paulet et al.'s scheme [25], and Zhu et al.'s scheme [26], respectively. Figure 4b shows the correlation between the computation cost of FN and the number of segments m, we can see that the computation cost of FN in the EP^{2}DS scheme is the smallest compared with Kong et al.'s scheme [23], Paulet et al.'s scheme [25], and Zhu et al.'s scheme [26]. The computation costs of FN in the proposed EP^{2}DS scheme are 3.0808 ms, which decreases by 88.6% and 93.4% compared with Kong et al.'s scheme [23] and Zhu et al.'s scheme [26]. Furthermore, unlike Paulet et al.'s scheme [25], the computation cost of FN in the EP^{2}DS scheme does not increase with the number of segments m.

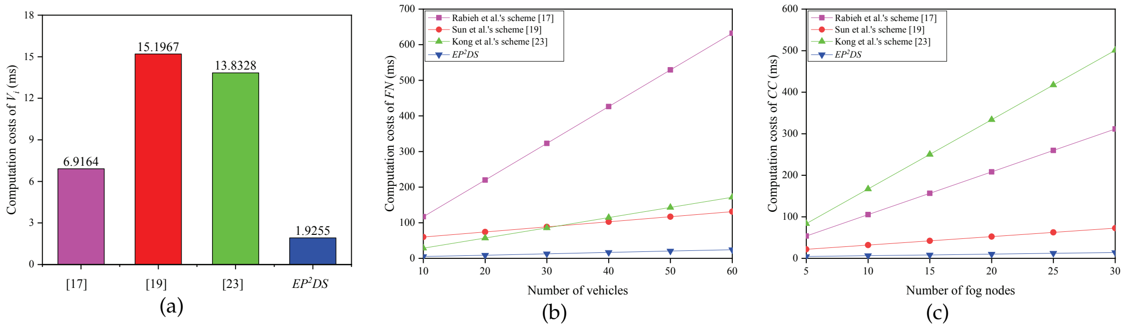


Figure 3. Computation costs in the data collection phase. (**a**) Computation costs of V_i; (**b**) Computation costs of FN vs. number of vehicles; (**c**) Computation costs of CC vs. number of FN.

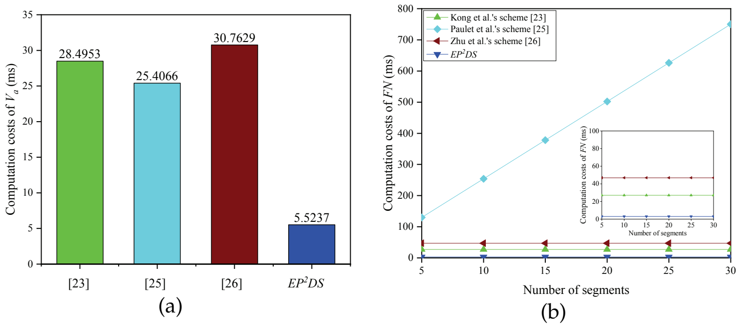


Figure 4. Computation costs in the data query phase. (**a**) Computation costs of V_q; (**b**) Computation costs of FN vs. number of segments.

6.2. Communication Costs

The communication costs of the proposed EP^{2}DS scheme and these schemes [17,19,23,25,26], are evaluated in this subsection. We mainly consider the data report size, query report size, and response report size. As mentioned above, the lengths of the elements in $\mathbb{G}$, $\mathbb{Z}_q^*$, $\mathbb{Z}_n$, and $\mathbb{Z}_{n^2}$ are 160 bits (20 bytes), 160 bits (20 bytes), 1024 bits (128 bytes), and 2048 bits (256 bytes), respectively, assuming that the length of timestamp and identity are 32 bits (4 bytes). The comparison results of communication costs are illustrated in Table 5.

Table 5. Comparison of the communication costs.

Scheme	Data Collection Phase	Data Query Phase	
	Data Report Size	Query Report Size	Response Report Size
Rabieh et al.'s scheme [17]	260 bytes	–	–
Sun et al.'s scheme [19]	516 bytes	–	–
Kong et al.'s scheme [23]	1152 bytes	1152 bytes	1664 bytes
Paulet et al.'s scheme [25]	–	256 bytes	256m+128 bytes
Zhu et al.'s scheme [26]	–	324 bytes	320 bytes
EP^{2}DS	172 bytes	172 bytes	148 bytes

In the data collection phase, for Rabieh et al.'s scheme [17], the data report size is 260 bytes, as

$$|C_v| + |TS| + |\alpha_v| = 128 + 4 + 128 = 260 \text{ bytes}.$$

For Sun et al.'s scheme [19], the data report size is 516 bytes, as

$$|S_c| + |SignC_i| + |t_i| = 256 + 256 + 4 = 516 \text{ bytes}.$$

For Kong et al.'s scheme [23], the data report size is 1152 bytes, as

$$|C_{i,1}| + |C_{i,2}| + |C_{i,3}| + |C_{i,4}| + |MAC_i| = 256 + 256 + 256 + 256 + 128 = 1152 \text{ bytes}.$$

For the proposed EP^{2}DS scheme, the data report size is 172 bytes, as

$$\begin{gathered}|PID_i| + |R_i| + |A_i^j| + |B_i^j| + |C_i^j| + |D_i^j| + |L_i^j| + |\sigma_i^j| + |T_i^j| \\ = 28 + 20 + 20 + 20 + 20 + 20 + 20 + 20 + 4 + 4 = 172 \text{ bytes}.\end{gathered}$$

In the data query phase, for Kong et al.'s scheme [23], the query report size is 1152 bytes, as

$$|C_{a,1}| + |C_{a,2}| + |C_{a,3}| + |C_{a,4}| + |MAC_a| = 256 + 256 + 256 + 256 + 128 = 1152 \text{ bytes}.$$

The response report size is 1664 bytes, as

$$\begin{gathered}|C_{r,1}| + |C_{r,2}| + |C_{r,3}| + |C_{r,4}| + |C_{r,5}| + |C_{r,6}| + |MAC_r| \\ = 256 + 256 + 256 + 256 + 256 + 256 + 128 = 1664 \text{ bytes}.\end{gathered}$$

For Paulet et al.'s scheme [25], the query report size is 256 bytes, as

$$|C_1| + |C_2| = 128 + 128 = 256 \text{ bytes}.$$

The response report size is 256m+128 bytes, as

$$\begin{gathered}|C'_{1,1}| + |C'_{1,2}| + \cdots + |C'_{1,m}| + |C'_{2,1}| + |C'_{2,2}| \cdots + |C'_{2,m}| + |\gamma| \\ = 128m + 128m + 128 = 256m + 128 \text{ bytes}.\end{gathered}$$

For Zhu et al.'s scheme [26], the query report size is 324 bytes, as

$$|ID_{LBS}| + |E_{LQR}| + |U_i| + |TS| + |Sig_i| = 4 + 256 + 256 + 4 + 256 = 324 \text{ bytes}.$$

The response report size is 320 bytes, as

$$|E_{rq_1}(TRL)| + |ID_{cs}| + |TS| + |Sig_{cs}| = 256 + 4 + 4 + 256 = 320 \text{ bytes}.$$

For the proposed EP^{2}DS scheme, the query report size is 172 bytes, as

$$\begin{aligned} &|PID_q| + |R_q| + |A_q^j| + |B_q^j| + |C_q^j| + |D_q^j| + |L_q^j| + |\sigma_q^j| + |T_q^j| \\ = {} & 28 + 20 + 20 + 20 + 20 + 20 + 20 + 20 + 4 = 172 \text{ bytes}. \end{aligned}$$

The response report size is 148 bytes, as

$$\begin{aligned} &|ID_{FN_j}| + |R_{FN_j}| + |J_q^j| + |K_q^j| + |M_q^j| + |N_q^j| + |\hat{L}_q^j| + |\hat{\sigma}_q^j| + |\hat{T}_q^j| \\ = {} & 4 + 20 + 20 + 20 + 20 + 20 + 20 + 20 + 4 = 148 \text{ bytes}. \end{aligned}$$

The results from the comparison of communication costs in the data collection phase are illustrated in Figure 5. In terms of the data report size, the proposed EP^{2}DS scheme requires 172 bytes, which is decreased by 33.8%, 66.7%, and 85.1% compared with that for Rabieh et al.'s scheme [17], Sun et al.'s scheme [19], and Kong et al.'s scheme [23], respectively.

The result from the comparison of communication costs in the data query phase is shown in Figure 6. Regarding the query report size, from Figure 6a, we can see that the proposed EP^{2}DS scheme requires 172 bytes, a decrease of 85.1%, 32.8%, and 46.9% compared with that by Kong et al.'s scheme [23], Paulet et al.'s scheme [25], and Zhu et al.'s scheme [26], respectively. Figure 6b shows the correlation between the response report size and the number of segments m, and we can see that the response report size in the EP^{2}DS scheme is the smallest compared with Kong et al.'s scheme [23], Paulet et al.'s scheme [25], and Zhu et al.'s scheme [26]. The proposed EP^{2}DS scheme requires 148 bytes, which is decreased by 91.1% and 53.8% compared with that of Kong et al.'s scheme [23] and Zhu et al.'s scheme [26], respectively. Furthermore, unlike Paulet et al.'s scheme [25], the response report size in the EP^{2}DS scheme does not increase with the number of segments m.

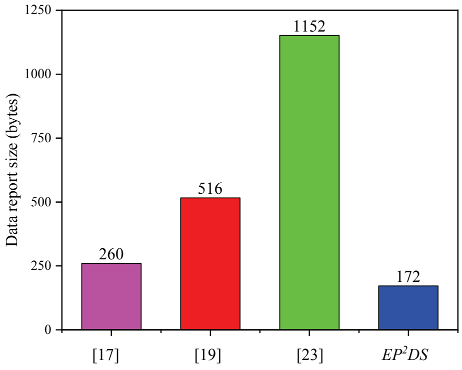
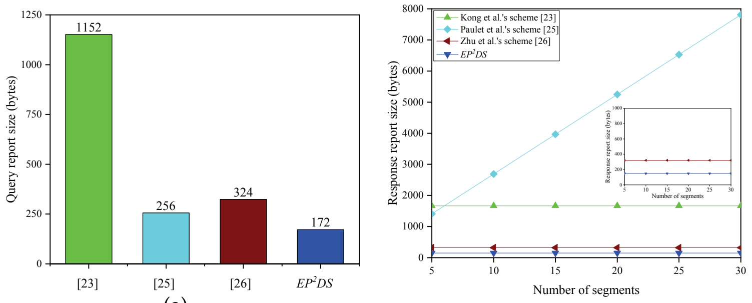


Figure 5. Comparison of the data report size.



(a) (b)

Figure 6. (**a**) Comparison of the query report size; (**b**) Comparison of the response report size.

7. Conclusion

This paper proposes an efficient privacy-preserving data sharing scheme for fog-assisted vehicular sensor networks. Based on the super-increasing sequence, the proposed EP^{2}DS scheme is able to format the data reports captured at different road segments into one report, while calculating the average sensory data in each road segment, greatly saving on the resources of communication and computation. Furthermore, by exploiting the modified oblivious transfer technology, the proposed EP^{2}DS scheme also can query the road conditions of the potential moving route in the data query phase without disclosing the query location. Finally, an analysis of security displays that the proposed EP^{2}DS scheme can satisfy all the requirements for security and privacy, with the performance evaluation suggesting that the proposed EP^{2}DS scheme is more efficient in computation and communication costs compared to the existing schemes of [17,19,23,25,26]. Accordingly, the proposed EP^{2}DS scheme is more appropriate for achieving data sharing in fog-assisted vehicular sensor networks. In future work, we will consider using blockchain technology to achieve decentralization and privacy protection.

Author Contributions: Y.M. and X.Y. conceived of the work, designed the concrete scheme, and wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Natural Science Foundation of Shaanxi Province under Grant 2018JM6081, and in part by the Fundamental Research Funds for the Central Universities, CHD, under Grant 300102249204.

Acknowledgments: The authors would like to thank the anonymous reviewers for their constructive comments and suggestions which helped us to improve the organization and presentation of this paper.

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

Appendix A

Security Model

The proposed EP^{2}DS scheme should satisfy the confidentiality and unforgeability. The security is defined by the following two interaction games executed by a challenger $\mathcal{C}$ and an attacker $\mathcal{A}$. $\mathcal{A}$ could make the following queries.

- **Hash queries**: Upon receiving the query, $\mathcal{C}$ returns a random value to $\mathcal{A}$.
- **Extract queries**: Upon receiving the query on the pseudo identity PID_i, $\mathcal{C}$ returns a private key to $\mathcal{A}$.
- **Signcryption queries**: Upon receiving the query on the message m_i under PID_i, $\mathcal{C}$ returns a ciphertext to $\mathcal{A}$.

Definition A1 (Confidentiality)**.** *The proposed scheme is secure against indistinguishability under the chosen plaintext attack (IND-CPA), if any probabilistic polynomial-time attacker does not have the ability to win the below game with a non-negligible advantage.*

The IND-CPA is defined by the following game.

Setup: $\mathcal{C}$ generates the system parameters and returns to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ adaptively makes the hash, extract, and signcryption queries with polynomial bounded times.

Challenge: $\mathcal{A}$ chooses a challenging identity PID_i^*, picks two messages m_0^* and m_1^* and sends to $\mathcal{C}$. $\mathcal{C}$ randomly picks $b \in \{0,1\}$ and produces the ciphertext of message m_b^* under PID_i^*. Finally, $\mathcal{C}$ returns the ciphertext to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ is able to adaptively perform the query in Phase 1 apart from that, it cannot make extract queries on PID_i^*.

Guess: $\mathcal{A}$ produces a guess $b' \in \{0,1\}$. The advantage that $\mathcal{A}$ wins the game is

$$Adv_{\mathcal{A}}^{IND-CPA} = |\Pr[b' = b] - \tfrac{1}{2}|.$$

Definition A2 (Unforgeability)**.** *The proposed scheme can achieve existential unforgeability against adaptive chosen message attacks (EUF-CMA), if any probabilistic polynomial-time attacker does not have the ability to win the below game with a non-negligible advantage.*

The EUF-CMA is defined by the following game.
Initialization: $\mathcal{A}$ selects a challenging pseudo identity PID_i^* and transmits to $\mathcal{C}$.
Setup: $\mathcal{C}$ generates the system parameters and returns to $\mathcal{A}$.
Queries: $\mathcal{A}$ adaptively makes hash, extract and signcryption queries.
Forgery: $\mathcal{A}$ outputs a ciphertext on m_i^* under PID_i^*, such that

- The ciphertext on m_i^* under PID_i^* is valid.
- PID_i^* has not been requested in the extract queries.

Appendix B

Security Proof

Theorem A1. *The proposed EP^{2}DS scheme can provide confidentiality if ElGamal encryption is secure against the IND-CPA.*

Supposing there is an attacker $\mathcal{A}$ is able to win the game defined in Definition 1 with a non-negligible probability ε, we can construct an algorithm $\mathcal{B}$ that could break the *IND-CPA* of ElGamal encryption with probability ε'.

Initialization: The simulator $\mathcal{S}$ for ElGamal encryption generates the $\{p, q, P, \mathbb{G}, P_{pub})$ and transmits to $\mathcal{B}$.

Setup: $\mathcal{B}$ chooses hash functions H_i: $i = 1, 2, \cdots, 8$ and a super-increasing sequence $\vec{a}$. Finally, $\mathcal{B}$ returns $\{p, q, P, \mathbb{G}, P_{pub}, P_{sp}, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, \vec{a}\}$ to $\mathcal{A}$.

To keep the rapidly response and consistency, $\mathcal{B}$ maintains the following list:

- L_{H_2}: It consists of tuples $(PID_i, R_i, P_{pub}, h_i)$.
- L_{H_4}: It consists of tuples $(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i, \tau_i)$.
- L_{V_i}: It consists of tuples (PID_i, x_i, R_i).

Phase 1: $\mathcal{A}$ adaptively is able to adaptively perform the following polynomial bounded times queries.

H_2 **queries**: $\mathcal{A}$ performs a query on (PID_i, R_i, P_{pub}), $\mathcal{B}$ executes as follows:

- If L_{H_2} contains $(PID_i, R_i, P_{pub}, h_i)$, $\mathcal{B}$ responds with the previous value $h_i = H_2(PID_i, R_i, P_{pub})$ to $\mathcal{A}$.
- If L_{H_2} does not contain $(PID_i, R_i, P_{pub}, h_i)$, $\mathcal{B}$ randomly chooses a number $h_i \in \mathbb{Z}_q^*$, adds $(PID_i, R_i, P_{pub}, h_i)$ into L_{H_2} and returns h_i to $\mathcal{A}$.

H_4 **queries**: $\mathcal{A}$ performs a query on $(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i)$, $\mathcal{B}$ executes as follows:

- If L_{H_4} contains $(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i, \tau_i)$, $\mathcal{B}$ responds with the previous value $\tau_i = H_4(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i)$ to $\mathcal{A}$.
- If L_{H_4} does not contain $(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i, \tau_i)$, $\mathcal{B}$ randomly chooses a number $\tau_i \in \mathbb{Z}_q^*$, adds $(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i, \tau_i)$ into L_{H_4} and returns τ_i to $\mathcal{A}$.

Extract queries: $\mathcal{A}$ performs a query on PID_i, $\mathcal{B}$ executes as follows:

- If $PID_i = PID_i^*$, $\mathcal{B}$ aborts the game.
- If $PID_i \neq PID_i^*$, $\mathcal{B}$ executes:

  - If L_{V_i} contains (PID_i, x_i, R_i), $\mathcal{B}$ returns (x_i, R_i) to $\mathcal{A}$.

- If L_{V_i} does not contain (PID_i, x_i, R_i), $\mathcal{B}$ randomly chooses $x_i, h_i \in \mathbb{Z}_q^*$ and makes $R_i = x_iP - h_iP_{pub}$. If h_i already appear in L_{H_2}, $\mathcal{B}$ chooses another $x_i \in \mathbb{Z}_q^*$ and tries again. $\mathcal{B}$ inserts (PID_i, x_i, R_i) and $(PID_i, R_i, P_{pub}, h_i)$ into L_{V_i} and L_{H_2}, respectively. Finally, $\mathcal{B}$ returns the (x_i, R_i) to $\mathcal{A}$.

Signcryption queries: $\mathcal{A}$ makes a query on the message m_i under PID_i, $\mathcal{B}$ returns m_i to $\mathcal{S}$. $\mathcal{S}$ randomly chooses $t_i \in \mathbb{Z}_q^*$ and computes $C_{i,1} = t_iP$, $C_{i,2} = t_iP_{cc} + m_iP$,and returns them to $\mathcal{B}$. $\mathcal{B}$ produces a ciphertext $\{PID_i, R_i, C_{i,1}, C_{i,2}, L_i, \sigma_i, T_i\}$ in accordance with the proposed scheme. Finally, $\mathcal{B}$ returns the ciphertext to $\mathcal{A}$.

Challenge: $\mathcal{A}$ selects a challenging identity PID_i^*, picks two same length message m_0^* and m_1^* and sends them to $\mathcal{B}$. Then $\mathcal{B}$ transmits them to $\mathcal{S}$. $\mathcal{S}$ randomly chooses $b \in \{0, 1\}$, $t_i^* \in \mathbb{Z}_q^*$ and computes $C_{i,1}^* = t_i^*P$, $C_{i,2}^* = t_i^*P_{cc} + m_b^*P$, and returns them to $\mathcal{B}$. $\mathcal{B}$ produce a ciphertext $\{PID_i^*, R_i^*, C_{i,1}^*, C_{i,2}^*, L_i^*, \sigma_i^*, T_i^*\}$ in accordance with the proposed scheme. Finally, $\mathcal{B}$ returns the ciphertext to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ is able to adaptively perform the query in Phase 1 apart from it cannot make a extract queries on PID_i^*.

Guess: $\mathcal{B}$ can output b' as its guess against the *IND-CPA* of ElGamal encryption.

Probability analysis: Supposing that $\mathcal{A}$ is able to make at most q_{H_2} times H_2 queries, q_{H_4} times H_4 queries, q_e times extract queries and q_s times signcryption queries. We define two events as follows:

- E_1: $\mathcal{B}$ does not abort above game in extract queries.
- E_2: $\mathcal{B}$ is able to correctly output the value of b.

According to the above simulation, we could obtain that $\Pr[E_1] \geq (1 - \frac{1}{q_{H_2}})^{q_e}$ and $\Pr[E_2|E_1] \geq \varepsilon$, and hence the advantage that $\mathcal{B}$ is able to break the *IND-CPA* of ElGamal encryption is

$$\varepsilon' = \Pr[E_2|E_1]\Pr[E_1] \geq (1 - \frac{1}{q_{H_2}})^{q_e}\varepsilon.$$

In accordance with the above analysis, we can conclude that $\mathcal{B}$ can break the *IND-CPA* of ElGamal encryption with a non-negligible probability, this is contradicts with the security of ElGamal encryption, so the proposed EP^{2}DS scheme could provide confidentiality.

Theorem A2. *The proposed EP^{2}DS scheme can provide the unforgeability if the ECDL problem is hard.*

Assuming that there is an attacker $\mathcal{A}$ can break the unforgeability of the proposed EP^{2}DS scheme with a non-negligible advantage ε, we can construct an algorithm $\mathcal{B}$ for solving the ECDL problem with probability ε'.

Initialization: $\mathcal{A}$ picks a challenging identity PID_i^* and returns to $\mathcal{B}$.

Setup: Given an instance $(P, aP = Q)$ of the ECDL problem, then $\mathcal{B}$ sets $P_{pub} = Q$ and returns $\{p, q, P, \mathbb{G}, P_{pub}, P_{sp}, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, \vec{a}\}$ to $\mathcal{A}$.

H_2 **queries**: It is the same as Theorem 1.

H_4 **queries**: It is the same as Theorem 1.

Extract queries: It is the same as Theorem 1.

Signcryption queries: $\mathcal{A}$ makes a query on the message m_i under PID_i, $\mathcal{B}$ executes as follows:

- If $PID_i = PID_i^*$, $\mathcal{B}$ randomly selects $t_i, l_i, \sigma_i, h_i, \tau_i \in \mathbb{Z}_q^*$ and calculates $C_{i,1} = t_iP$, $C_{i,2} = t_iP_{cc} + m_iP$, $L_i = l_iP$, $R_i = \sigma_iP - (h_iP_{pub} + \tau_iL_i)$. If the h_i already appears in L_{H_2} or τ_i already appears in L_{H_4}, $\mathcal{B}$ chooses another $\sigma_i \in \mathbb{Z}_q^*$ and tries again. Then, $\mathcal{B}$ returns the ciphertext $\{PID_i, R_i, C_{i,1}, C_{i,2}, L_i, \sigma_i, T_i\}$ to $\mathcal{A}$, and inserts $(PID_i, R_i, P_{pub}, h_i)$ and $(PID_i, R_i, C_{i,1}, C_{i,2}, L_i, T_i, \tau_i)$ into L_{H_2} and L_{H_4}, respectively.
- If $PID_i \neq PID_i^*$, $\mathcal{B}$ generates a ciphertext $\{PID_i, R_i, C_{i,1}, C_{i,2}, L_i, \sigma_i, T_i\}$ in accordance with the proposed scheme. Then, $\mathcal{B}$ returns the ciphertext to $\mathcal{A}$.

Forgery: $\mathcal{A}$ outputs a forged ciphertexts $\{PID_i^*, R_i^*, C_{i,1}^*, C_{i,2}^*, L_i^*, \sigma_i^*, T_i^*\}$ on m_i^* under PID_i^*. On the basis of the forking lemma [40,41], $\mathcal{B}$ is able to output another valid ciphertext $\{PID_i^*, R_i^*, C_{i,1}^*, C_{i,2}^*, L_i^*, \sigma_i^{*'}, T_i^*\}$ on m_i^* under PID_i^* by choosing a different H_2. Since both ciphertexts are valid, we are able to gain the following two equations

$$\sigma_i^* P = R_i^* + h_i^* P_{pub} + \tau_i^* L_i, \sigma_i^{*'} P = R_i^* + h_i^{*'} P_{pub} + \tau_i^* L_i.$$

We can gain the equations:

$$(\sigma_i^* - \sigma_i^{*'})P = \sigma_i^* P - \sigma_i^{*'} P = (h_i^* - h_i^{*'})P_{pub} = (h_i^* - h_i^{*'})aP.$$

$\mathcal{B}$ outputs $a = (h_i^* - h_i^{*'})^{-1}(\sigma_i^* - \sigma_i^{*'})$ as a solution of ECDL problem.

Probability analysis: Supposing that $\mathcal{A}$ is able to make at most q_{H_2} times H_2 queries, q_{H_4} times H_4 queries, q_e times extract queries, and q_s times signcryption queries. We define three events as follows:

- E_1: $\mathcal{B}$ never abort above game in extract and signcryption queries.
- E_2: $\mathcal{B}$ is able to output a valid ciphertext.
- E_3: $PID_i = PID_i^*$.

According to the above simulation, we could obtain that $\Pr[E_1] \geq (1 - \frac{1}{q_{H_2}})^{q_e}(1 - \frac{1}{q_{H_4}})^{q_s}$, $\Pr[E_2|E_1] \geq \varepsilon$, and $\Pr[E_3|E_1 \wedge E_2] \geq \frac{1}{q_{H_2}}$. Thus, the probability that $\mathcal{B}$ is able to solve the ECDL problem is shown as:

$$\varepsilon' = \Pr[E_1 \wedge E_2 \wedge E_3] \geq \Pr[E_3|E_1 \wedge E_2] \Pr[E_2|E_1] \Pr[E_1] \geq \frac{1}{q_{H_2}}(1 - \frac{1}{q_{H_2}})^{q_e}(1 - \frac{1}{q_{H_4}})^{q_s}\varepsilon.$$

Due to the non-negligibility of ε, we are able to know that ε' is non-negligible. In accordance with the above analysis, we are able to conclude that $\mathcal{B}$ can solve the ECDL problem with a non-negligible probability. This contradicts with the hardness of the ECDL problem [42], and hence the proposed EP^{2}DS scheme can provide unforgeability.

References

1. Lee, U.; Magistretti, E.; Zhou, B.; Gerla, M.; Bellavista, P.; Corradi, A. MobEyes: Smart mobs for urban monitoring with a vehicular sensor network. *IEEE Trans. Commun. Mag.* **2006**, *13*, 52–57. [CrossRef]
2. Placzek, B. Selective data collection in vehicular networks for traffic control applications. *Transp. Res. Part C: Emerging Technol.* **2012**, *23*, 14–28. [CrossRef]
3. Mednis, A.; Elsts, A.; Selavo, L. Embedded solution for road condition monitoring using vehicular sensor networks. In Proceedings of the 2012 6th International Conference on Application of Information and Communication Technologies (AICT), Tbilisi, Georgia, 17–19 October 2012; pp. 1–5.
4. Fiebig, B. European traffic accidents and purposed solutions. In Proceedings of the ITU-Workshop on Standardization in Telecommunication for Motor Vehicles, Geneva, Switzerland, 24–25 November 2003; pp. 24–25.
5. Yu, R.; Huang, X.; Kang, J.; Ding, J.; Maharjan, S.; Gjessing, S.; Zhang, Y. Cooperative resource management in cloud-enabled vehicular networks. *IEEE Trans. Ind. Electron.* **2015**, *62*, 7938–7951. [CrossRef]
6. Ni, J.; Lin, X.; Zhang, K.; Shen, X.M. Privacy-preserving real-time navigation system using vehicular crowdsourcing. In Proceedings of the IEEE 84th Vehicular Technology Conference: VTC2016-Fall, Montreal, QC, Canada, 18–21 September 2016; pp. 1–5.
7. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog computing and its role in the internet of things. In Proceedings of the Mobile Cloud Computing Workshop, Helsinki, Finland, 13–17 August 2012; pp. 13–16.
8. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.H.; Konwinski, A.; Lee, G.; Patterson, D.A.; Rabkin, A.; Stoica, I.; et al. A view of cloud computing. *Commun. ACM* **2010**, *53*, 50–58. [CrossRef]
9. Dai, Y.; Xu, D.; Maharjan, S.; Zhang, Y. Joint offloading and resource allocation in vehicular edge computing and networks. In Proceedings of the IEEE Global Communications Conference, Abu Dhabi, UAE, 9–13 December 2018; pp. 1–7.

10. Ni, J.; Zhang, K.; Yu, Y.; Lin, X.; Shen, X.S. Privacy-preserving smart parking navigation supporting efficient driving guidance retrieval. *IEEE Trans. Veh. Technol.* **2018**, *67*, 6504–6517. [CrossRef]
11. Basudan, S.; Lin, X.; Sankaranarayanan, K. A privacy-preserving vehicular crowdsensing based road surface condition monitoring system using fog computing. *IEEE Internet Things J.* **2017**, *4*, 772–782. [CrossRef]
12. Chun, S.; Shin, S.; Seo, S.; Eom, S.; Jung, J.; Lee, K. A pubsub-based fog computing architecture for Internet-of-vehicles. In Proceedings of the 8th International Conference on Cloud Computing Technology and Science, Luxembourg, 12–15 December 2016; pp. 90–93.
13. Ni, J.; Zhang, A.; Lin, X.; Shen, X.S. Security, privacy, and fairness in fog-based vehicular crowdsensing. *IEEE Commun. Mag.* **2017**, *55*, 146–152. [CrossRef]
14. Wei, J.; Wang, X.; Li, N. A privacy-preserving fog computing framework for vehicular crowdsensing betworks. *IEEE Access* **2018**, *6*, 43776–43784. [CrossRef]
15. Omoniwa, B.; Hussain, R.; Javed, M.A. Fog/Edge computing-based IoT (FECIoT): Architecture, applications, and research issues. *IEEE Internet Things* **2019**, *6*, 4118–4149. [CrossRef]
16. Zhuo, G.; Jia, Q.; Guo, L.; Li, M.; Li, P. Privacy-preserving verifiable data aggregation and analysis for cloud-assisted mobile crowdsourcing. In Proceedings of the 35th IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–14 April 2016; pp. 1–9.
17. Rabieh, K.; Mahmoud, M.M.E.A.; Younis, M. Privacy-preserving route reporting schemes for traffic management systems. *IEEE Trans. Veh. Technol.* **2017**, *66*, 2703–2713. [CrossRef]
18. Xu, C.; Lu, R.; Wang, H.; Zhu, L.; Huang, C. PAVS: A new privacy-preserving data aggregation scheme for vehicle sensing systems. *Sensors* **2017**, *17*, 1–18. [CrossRef] [PubMed]
19. Sun, G.; Sun, S.; Sun, J.; Yu, H.; Du, X.; Guizani, M. Security and privacy preservation in fog-based crowd sensing on the internet of vehicles. *J. Network Comput. Appl.* **2019**, *134*, 89–99. [CrossRef]
20. Lin, X.; Lu, R.; Shen, X. MDPA: Multidimensional privacy-preserving aggregation scheme for wireless sensor networks. *Wirel. Commun. Mob. Comput.* **2010**, *10*, 843–856. [CrossRef]
21. Lu, R.; Heung, K.; Lashkari, A.H.; Ghorbani, A.A. A light-weight privacy-preserving data aggregation scheme for fog computing-enhanced IoT. *IEEE Access* **2017**, *5*, 3302–3312. [CrossRef]
22. Wang, B.; Chang, Z.; Zhou, Z.; Ristaniemi, T. Reliable and privacy-preserving task recomposition for crowdsensing in vehicular fog computing. In Proceedings of the 87th Vehicular Technology Conference, Porto, Portugal, 3–6 June 2018; pp. 6–11.
23. Kong, Q.; Lu, R.; Ma, M.; Bao, H. A privacy-preserving sensory data sharing scheme in internet of vehicles. *Future Gener. Comput. Syst.* **2019**, *92*, 644–655. [CrossRef]
24. Ghinita, G.; Kalnis, P.; Kantarcioglu, M.; Bertino, E. A hybrid technique for private location-based queries with database protection. In Proceedings of the 11th International Symposium on Spatial and Temporal Databases, Aalborg, Denmark, 8–10 July 2009; pp. 98–116.
25. Paulet, R.; Kaosar, M.G.; Yi, X.; Bertino, E. Privacy-preserving and content protecting location based queries. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1200–1210. [CrossRef]
26. Zhu, H.; Lu, R.; Huang, C.; Chen, L.; Li, H. An efficient privacy-preserving location-based services query scheme in outsourced cloud. *IEEE Trans. Veh. Technol.* **2016**, *65*, 7729–7739. [CrossRef]
27. Zhu, H.; Liu, F.; Li, H. Efficient and privacy-preserving polygons spatial query framework for location-based services. *IEEE Internet Things J.* **2017**, *4*, 536–545. [CrossRef]
28. Naor, M.; Pinkas, B. Oblivious transfer with adaptive queries. In Proceedings of the Advances in Cryptology-CRYPTO'99. Santa Barbara, CA, USA, 15–19 August 1999; pp. 573–590.
29. IEEE, 802.11p-2010-IEEE Standard for Information technology. Available online: https://ieeexplore.ieee.org/document/5514475/versions#versions (accessed on 14 January 2020).
30. Jiang, S.; Liu, J.; Duan, M.; Wang, L.; Fang, L. Secure and privacy-preserving report de-duplication in the fog-based vehicular crowdsensing system. In Proceedings of the IEEE Global Communications Conference, Abu Dhabi, UAE, 9–13 December 2018; pp. 1–6.
31. Zhu, L.; Li, M.; Zhang, Z. Secure fog-assisted crowdsensing with collusion resistance: From data reporting to data requesting. *IEEE Internet Things J.* **2019**, *6*, 5473–5484. [CrossRef]
32. Miller, V.S. Use of elliptic curves in cryptography. In Proceedings of the Advances in Cryptology-CRYPTO'85, Santa Barbara, CA, USA, 18–22 August 1985; pp. 417–426.
33. Koblitz, N. Elliptic curve cryptosystem. *Math. Comput.* **1987**, *48*, 203–209. [CrossRef]

34. Ming, Y.; Zhang, X.; Shen, X. Efficient privacy-preserving multi-dimensional data aggregation scheme in smart grid. *IEEE Access* **2019**, *7*, 32907–32921. [CrossRef]
35. He, D.; Zeadally, S.; Xu, B.; Huang, X. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2681–2690. [CrossRef]
36. Liu, J.K.; Yuen, T.H.; Au, M.H.; Susilo, W. Improvements on an authentication scheme for vehicular sensor networks. *Expert Syst. Appl.* **2014**, *41*, 2559–2564. [CrossRef]
37. Boneh, D.; Goh, E.; Nissim, K. Evaluating 2-DNF formulas on ciphertexts. In Proceedings of the 2nd Theory of Cryptography Conference, Cambridge, MA, USA, 10–12 February 2005; pp. 325–341.
38. Ming, Y.; Cheng, H. Efficient certificateless conditional privacy-preserving authentication scheme in VANETs. *Mob. Inf. Syst.* **2019**, *2019*, 1–19. [CrossRef]
39. Shamus Software Ltd. Multi precision integer and rational arithmetic cryptographic library (MIRACL). Available online: http://www.certivox.com/miracl/ (accessed on 1 December 2019).
40. Pointcheval, D.; Stern, J. Security proofs for signature schemes. In Proceedings of the Advances in Cryptology-EUROCRYPT'96, Saragossa, Spain, 12–16 May 1996; pp. 387–398.
41. Ming, Y.; Shen, X. PCPA: A practical certificateless conditional privacy preserving authentication scheme for vehicular ad hoc networks. *Sensors*, **2018**, *18*, 1–23. [CrossRef] [PubMed]
42. He, D.; Kumar, N.; Zeadally, S.; Vinel, A.; Yang, L.T. Efficient and privacy-preserving data aggregation scheme for smart grid against internal adversaries. *IEEE Trans. Smart Grid* **2017**, *13*, 1–9. [CrossRef]