Software Defined Networking with Pseudonym Systems for Secure Vehicular Clouds

Xumin Huang, Rong Yu, Member, IEEE, Jiawen Kang, Ning Wang, Member, IEEE, Sabita Maharjan, Member, IEEE, and Yan Zhang (corresponding author), Senior Member, IEEE

Abstract—The vehicular cloud is a promising new paradigm where vehicular networking and mobile cloud computing are elaborately integrated to enhance the quality of vehicular information services. Pseudonym is a resource for vehicles to protect their location privacy, which should be efficiently utilized to secure vehicular clouds. However, only a few existing architectures of pseudonym systems take flexibility and efficiency into consideration, thus leading to potential threats to location privacy. In this paper, we exploit software-defined networking technology to significantly extend the flexibility and programmability for pseudonym management in vehicular clouds. We propose a software-defined pseudonym system where the distributed pseudonym pools are promptly scheduled and elastically managed in a hierarchical manner. In order to decrease the system overhead due to the cost of inter-pool communications, we leverage the two-sided matching theory to formulate and solve the pseudonym resource scheduling. We conducted extensive simulations based on the real map of San Francisco. Numerical results indicate that the proposed software-defined pseudonym system significantly improves the pseudonym resource utilization, and meanwhile, effectively enhances the vehicles’ location privacy by raising their entropy.

I. INTRODUCTION

With the rapid development of wireless communication technologies [1], [2], vehicles can utilize vehicle-to-infrastructure and vehicle-to-vehicle communications with the help of on-board devices to form vehicular networks. However, many emerging mobile applications require larger and secure storage [3] and complex computation, and brings new resource challenges to vehicular networks, e.g., vehicle platoon [4], real-time video streaming application [5]–[8] and vehicular augmented reality, social media sharing [9], [10]. To meet the growing demands of radio and computing resources, vehicular networks take the advantages of cloud computing and are evolving towards vehicular clouds. From a system-level view, idle resources in vehicles, network infrastructures (e.g., roadside unit (RSU)) and cloud infrastructures (e.g., data center) can be recruited to form a vehicular cloud system. A typical vehicular cloud system [11] consists of three different levels as following. 1) At the bottom level, cooperative vehicles create a vehicular cloud. 2) At the middle layer, a set of adjacent RSUs form a local cloud. 3) At the top layer, central cloud manages resources in the system. While ubiquitous wireless communication of pervasive cloud computing greatly facilitate the formation and functioning of vehicular cloud, privacy and security challenges remain to be addressed for this new domain [12], [13].

To secure vehicular clouds, we focus on pseudonym, which is an essential resource for vehicles to protect location privacy [14]. Most of the privacy protection schemes are implemented on the basis of pseudonyms, e.g., group signature, silent period, and mix-zone [14]. Vehicles should periodically change their pseudonyms to avoid being continuously tracked. Moreover, a third-party cloud service provider may pose potential threats to the vehicles because of data leakage [15]. This further highlights the importance of pseudonyms for vehicles to protect privacy in vehicular clouds. Vehicles need to possess sufficient pseudonyms to be able to frequently change for anonymity.

Moreover, with the increasing number of vehicles, pseudonym management in vehicular clouds has become a challenging problem. The drawbacks of a previous centralized approach to manage pseudonyms mainly include two aspects: a heavy computing workload for the central cloud and a big backhaul delay for the vehicles. These vulnerabilities confine the pseudonym system capacity, and also result in low utilization of pseudonyms. Consequently, the pseudonyms may not be sufficient to maintain the location privacy of the vehicles. To this end, a new pseudonym system with high flexibility and efficient pseudonym utilization is necessary. We exploit Software Defined Networking (SDN) to significantly enhance the flexibility and programmability for pseudonym management in vehicular clouds. Software defined networking is a novel technology to control the network in a logically centralized, programmable and systematic approach by decoupling the physical data plane and the abstract control plane [16]. The potential of centralized knowledge, programmability and flexibility in an SDN can satisfy the requirements of vehicular clouds and simplify pseudonym management, especially when the number of vehicles is high.

In this paper, we propose a Software-Defined Pseudonym System (SDPS), where distributed pseudonym pools are deployed, quickly scheduled and elastically managed in a hierarchical manner. Besides, to decrease the system overhead due to inter-pool communication, we leverage the two-sided matching theory to formulate and solve the pseudonym resource scheduling. The main contributions of this paper are summarized as follows.

- We propose a software-defined pseudonym system with
a hierarchical architecture, which leverages the SDN technology to provide flexibility and programmability for pseudonym management.

- We develop the two-sided matching theory to solve pseudonym resource scheduling problem, which matches the optimal pseudonym transmitters and receivers to decrease the system overhead due to inter-pool communication.
- Numerical results show that the proposed software-defined pseudonym system significantly improves the pseudonym resource utilization, and effectively strengthens the vehicles’ location privacy.

The rest of this paper is organized as follows. Section II presents the related work. We describe a new observation about delay on pseudonym distribution approaches in Section III. A hierarchical architecture of software-defined pseudonym system is proposed in Section IV. Section V discusses the pseudonym-allocation problem, and we introduce the two-sided matching theory to solve this problem in Section VI. Performance evaluation of our proposed scheme is provided in Section VII. Finally, Section VIII concludes this paper.

II. RELATED WORK

Recently, a few studies have investigated the combination of cloud computing and vehicular networks. The authors in [11] presented a hierarchical architecture to organize the cloud resources in a vehicular network, consisting of three layers: vehicular cloud, RSU cloud, and central cloud. In [17], the authors pointed out that the way of network service provisioning changes when integrating the mobile cloud model into vehicular networks. The Vehicular Ad hoc Networks (VANET) Cloud, a new cloud computing model for VANET as introduced in [18], consists of three layers: client layer, cloud layer and communication layer. [19] proposed a new two-tier BUS-VANET that enables less delivery delay and higher delivery rate than those of the traditional VANET.

Along with the system architectures and design principles, some researchers have shown great interest in the resource allocation problem in vehicular clouds. Due to uncertainty of the vehicles’ behavior, the variation of available computation resources in vehicular clouds cannot be neglected. To address this problem, the authors in [20] proposed an optimal computation resource allocation scheme. The dynamic vehicular clouds make a decision about whether or not to locally process a service request. Then the computing resource allocation problem in a vehicular cloud is formulated as a semi-Markov decision process to maximize the total long-term reward of the vehicles. The authors in [11] focused on resource allocation and formulated the competition among virtual machines as a non-cooperative game. Similarly, RSU cloud resource management models in [21] employed SDN technology to decrease virtual machine migration, and minimize the number of service hosts and the infrastructure routing delay.

SDN is emerged as a promising approach for providing a centralized control method for global resource management in cloud computing environment. The authors in [22] combined the SDN framework with cloud computing for cloud resource optimal control. A resource sharing strategy is designed with global optimum in the control plane and executed by each cloud service provider in the data plane. [23] exploited the SDN technology to allow the flexible allocation of bandwidth coordinated with virtual machine provisioning to minimize users costs. An optimal bandwidth provisioning and routing decision on virtualized routers are made by a SDN controller and then implemented on the physical network. Similar work on bandwidth allocation based on SDN was studied in [24] for guaranteeing quality of service. SDN bridges the gaps through unified network abstraction and programmability, which also can be utilized for overcoming today’s limitations in vehicular networks [25]. Through utilizing the SDN framework to manage the cloud resources in vehicular clouds, a new paradigm of 5G-enabled vehicular networks was proposed in [26]. With SDN technology reconfiguring resources, an efficient RSU cloud resource management scheme aiming to minimize reconfiguration overhead was proposed in [21]. In this paper, we also consider that SDN can be to coordinate among vehicles and allocate efficiently all kinds of resources in vehicular clouds.

Pseudonyms are crucial for vehicles to protect their location privacy when forming a vehicular cloud for inter-vehicular communication [14]. Vehicles need sufficient pseudonyms to frequently change for location privacy preservation. The schemes for pseudonym distribution can be broadly categorized into two groups. I) A centralized pseudonym pool distributes pseudonyms to vehicles. In [27], each vehicle obtains 48830 pseudonyms at a time, and uses these pseudonyms over a long time (e.g., one year). II) Distributed pseudonym pools distribute pseudonyms to vehicles by distributed pseudonym pools. In [28], the vehicles periodically obtain a certain number of resource (keys or pseudonyms) from local managers.

For efficient generation and management of pseudonyms, we adopt a distributed approach that distributed local cloud with a pseudonym pool generates and manages pseudonyms. This approach can reduce pseudonym distribution delay and balance the computing workload in vehicular clouds. To improve pseudonym utilization efficiency and to provide flexibility on pseudonym management, we exploit SDN and propose a new pseudonym system for vehicular clouds. The vehicles are mobile in both time and space, consequently causing different pseudonym demands in time and from different pseudonym pools. To address this issue, we design an efficient pseudonym scheduling and distribution scheme using the two-sided matching theory.

III. A NEW OBSERVATION ON PSEUDONYM DISTRIBUTION

In this section, we first introduce two pseudonym management approaches in detail. Furthermore, we make an observation about pseudonym distribution and find out the advantages of distributed pseudonym management approach.

A. Two Pseudonym Management approaches

In the centralized pseudonym management approach, a centralized pseudonym pool stores all pseudonyms and certificates, and distributes them to the vehicles for privacy
protection. Vehicles request and obtain pseudonyms through RSUs. All the vehicles send pseudonym requests with digital signatures to nearby RSUs after encryption. The RSUs decrypt and verify the pseudonym requests, and transmit these requests to the central manager after encrypting and adding signatures of the RSUs. The central manager decrypts and verifies the signatures generated by the RSUs and the vehicles. The central manager encrypts the pseudonyms and transmits them to the RSUs. After decryption and verification, the RSUs send the pseudonyms to the vehicles.

For distributed pseudonym-management, there is a local authority and a pseudonym pool in the local cloud. Vehicles request pseudonyms from the local clouds. The process of pseudonym distribution in the distributed approach is simpler. The local authorities generate and manage their pseudonyms in their own pseudonym pools. A vehicle sends an encrypted request with signature to its nearby RSU, which delivers the requests to a local authority. The local authority decrypts and verifies the request, and then distributes the encrypted pseudonyms to the vehicle. The vehicle verifies and receives the pseudonyms from the RSU. We observe that there are less handshake protocols and data transmission delay in the distributed approach. Besides, for central pseudonym management approach, all the pseudonyms include corresponding public and private keys and certificates. This brings a heavy computing workload to the central cloud from pseudonyms generation to revocation. A distributed pseudonym management approach can be helpful to balance this computing workload.

B. An experiment about Pseudonym Distribution

In this subsection, we compare the distribution delay of pseudonyms in different pseudonym management approaches. We select a map of the West University Place and Braeswood Place, Houston [29] as observation areas. Twelve RSUs are deployed in this map according to the scheme proposed in [30]. There are four local clouds in the experiment, each consisting of four adjacent RSUs. Some of the vehicles are mobile within the region of interest. We consider that the request for pseudonyms from the vehicles in different local clouds follows a Poisson process. The average key size is 1024 bits in RSA algorithm [31]. The time taken to execute basic operations in our experiment is referred from [32].

Fig. 1 shows that the distribution delay increases with the increase in average arrival rate of the vehicles that request pseudonyms. The pseudonyms distribution delay in the centralized approach is higher compared to the distributed approach. Moreover, it is clear that the computing overhead of basic operations of pseudonyms management (e.g., signing, encrypting and decrypting) in the centralized pseudonym management approach is higher than that in the distributed approach since there are more handshake protocols in the former. The central authority manages pseudonyms of all the vehicles, while the local authorities only manage a part of the vehicles. Therefore, the distributed approach is more efficient than the centralized approach because of smaller distribution delay and lower computation overhead.

IV. SOFTWARE-DEFINED PSEUDONYM SYSTEMS

In this section, we propose a software-defined pseudonym system, where distributed pseudonym pools are deployed, scheduled and elastically managed in a hierarchical manner.

A. SDN for Pseudonym Management

SDN has emerged as a novel approach to control the network in a centralized, programmable and systematic manner. The core concept of an SDN is the separation between the control plane and the data plane. By decoupling these two planes, network intelligence and state can be logically centralized and the data forwarding is abstracted from applications [33]. The flexibility of SDN can be an important advantage for cloud resource allocation to meet dynamic demands, and to improve resource utilization in vehicular clouds [21]. We exploit the SDN concept to increase the flexibility and programmability for pseudonym management in vehicular clouds. To deploy SDN, a communication protocol between the control plane and the data plane is required. We use the OpenFlow protocol, which is the de facto standard protocol for SDN. It consists of OpenFlow controller and OpenFlow switches. We design the pseudonym resource scheduling strategy in the control plane. Utilizing this strategy, the OpenFlow controller defines pseudonym forwarding rules for every OpenFlow switch in the pseudonym (data) plane. Some benefits of leveraging SDN in the context of pseudonym management are as follows.

- Globalization: The centralized controller obtains global knowledge about pseudonym resource, i.e., demand and consumption rates of all local clouds. With these information, an optimal resource scheduling strategy is designed to allocate the pseudonyms on demand efficiently.
- Flexibility: The SDN technology brings flexibility and programmability into the vehicular clouds for pseudonym management. Pseudonyms can be flexibly managed according to the heterogeneous characteristics of vehicular networks, such as mobility, topology and capability.
• Simplicity: By decoupling the pseudonym resource controls (control plane) and pseudonym forwarding functions (data plane), the SDN simplifies pseudonym management. This goal can be achieved even if the number of vehicles is high.

B. A Hierarchical Architecture for SDPS

Fig. 2 shows a hierarchical architecture for SDPS in vehicular clouds, which is divided into data plane and control plane. The vehicular clouds in this paper have three-layer clouds: central cloud, local cloud and vehicular cloud. There are a registration authority, a data center and an OpenFlow controller in the central cloud. The registration authority manages the digital certificates of all entities, e.g., vehicles, RSUs, OpenFlow switches, and pseudonym pools. The registration authority is in charge of monitoring the behaviors of all entities to ensure system security [27]. The data centers collect and store the status information of all local clouds. These information include traffic flow, and the deployment information of pseudonyms, which are used to design the optimal pseudonym resource scheduling strategy. Some adjacent RSUs and a remote data center form a local cloud, including a pseudonym pool with an OpenFlow switch. A group of cooperative vehicles create a vehicular cloud to share vehicular resources.

Pseudonyms are utilized in frequent vehicle-to-vehicle and vehicle-to-infrastructure communication for location privacy preservation. For example, when nearby vehicles in motion constitute a dynamic vehicular cloud, inter-vehicle communication is normally required. For location privacy preservation, the vehicles without sufficient pseudonyms send pseudonym requests to nearby RSUs. The local cloud schedules pseudonyms generated by its pseudonym pool to support the demands from vehicles. Generally, the pseudonym demands from vehicles in different local clouds may change over time. This means that there exists redundant or on-demand pseudonym resource among the local clouds.

In the SDPS, pseudonyms are generated by local pseudonym pools and transferred to other pseudonym pools in different local clouds when necessary. The pseudonyms are managed by the local clouds that distribute them. When some pseudonyms are distributed to a vehicle, these pseudonyms will be attached with signatures of the local clouds to indicate the manager. For example, a vehicle obtains some pseudonyms from the local cloud $LC_1$. $LC_1$ signs the pseudonyms and the vehicle may enter another local cloud, e.g., $LC_2$. $LC_2$ verifies the signatures of the pseudonyms to authenticate the vehicle. If the vehicle wants to request new pseudonyms from $LC_2$, $LC_2$ need to inform $LC_1$ to perform revocation of the former pseudonyms distributed to the vehicle. Then $LC_2$ distributes new pseudonyms to the vehicle.

The OpenFlow controller collects and analyzes the global status information in vehicular clouds. To improve pseudonym utilization, the global controller makes an optimal pseudonym resource scheduling strategy, and then OpenFlow switches forward pseudonym resource. A pseudonym-flow table is designed by the controller and sent to every OpenFlow switch. OpenFlow switches receive the pseudonym-flow table, and forward the pseudonyms to vehicles or other pseudonym pools according to the flow rules. The system consists of the following SDN components.

• OpenFlow controller: In the control plane, the OpenFlow controller is the logical central intelligence of the vehicular clouds, which controls the network behavior
of the entire system. The controller designs the optimal pseudonym resource scheduling strategy and generates a detailed pseudonym-flow table for every OpenFlow switch.

- **OpenFlow switch:** In the data plane, the pseudonym pools equipped with OpenFlow switches are controlled by the OpenFlow controller to perform actions. They are stationary elements of data plane, which are responsible for forwarding pseudonym-flow, e.g., forwarding pseudonyms to local vehicles or other pseudonym pools. More details about functions of data plane and control plane are shown in Fig. 3 and are described next.

- **Data plane:** The pseudonym pools in local clouds generate pseudonyms at a constant rate. There is an OpenFlow switch in every pseudonym pool, and every OpenFlow switch communicates with the OpenFlow controller. According to flow rules in a pseudonym-flow table designed by the OpenFlow controller, a pseudonym pool may distribute the pseudonyms to relative RSUs to make vehicles anonymous for privacy preservation in its coverage. On the other hand, it can also transmit redundant pseudonyms to others, or receive a certain number of pseudonyms from others. Therefore, the data plane is responsible for performing pseudonym-flow forwarding tasks in this system. Besides, status information about OpenFlow switches are also uploaded to the controller for checking.

- **Control plane:** The OpenFlow controller in the central cloud obtains global information about all the pseudonym pools and pseudonym requests from vehicles. The OpenFlow controller makes the optimal pseudonym resource allocation strategy among pseudonym pools. A pseudonym-flow table is also designed by the controller, and then it decides how the pseudonyms are forwarded in the vehicular clouds. The format of an item in a pseudonym-flow table is shown as: \(\text{PID} | \text{From} | \text{To} | \text{Time}\). Here, “PID” denotes the identification of pseudonym. “From” and “To” indicate where the pseudonym is generated from and transmitted to, respectively. “To” can be an address of an RSU or other pseudonym pools. “Time” is the timestamp of pseudonym generation. The goal of the pseudonym-flow table is to maximize the utilization of pseudonym resource by transmitting redundant pseudonyms to the pseudonym pools that fail short of pseudonyms. Due to the cost of inter-pools communication, the redundant pseudonyms should be well scheduled and transferred from pseudonym transmitters to receivers among the pseudonym pools. To efficiently match transmitters and receivers, we use two-sided matching theory to obtain the optimal result after multi-rounds matching.

V. Problem Formulation

In our model, the pseudonym pools with OpenFlow switches form a network as an undirected graph \(G = G(V, E)\). The network of the pseudonym pools includes \(m\) nodes (i.e., pseudonym pools) and \(n\) node pairs (i.e., edges and links). The pseudonym pools in local clouds are denoted by \(V = \{P_1, P_2, ..., P_m\}\). The set of edges \(E\) represents the undirected pseudonym transmission links. The pseudonym data packets can be transmitted between two connected pseudonym pools via wired link with smaller cost. During the transmission of pseudonym data packets, the data packet loss per distance unit is \(l\) [34]. Then the weights of edges are calculated by the total pseudonym transmission loss (denoted as \(c\)) between two connected pseudonym pools. Here, \(c = l \cdot d\), where \(d\) is the distance between two connected pseudonym pools. All the pseudonym pools are connected with each other. Using Dijkstra’s algorithm, the link with minimum communication cost between any two pseudonym pools can be determined. Defining a symmetric matrix \(M = \text{Dijkstra}(G)\) as the inter-pool minimum communication cost matrix, the element of the matrix, \(m_{i,j}(i \neq j)\), represents the minimum communication cost between pseudonym pool \(P_i\) and pseudonym pool \(P_j\). To make this paper clear, we use \(m(P_i, P_j)\) to replace \(m_{i,j}\).

At the beginning of an observation period \(t\) (i.e., a time window), a pseudonym pool \(P_i\) possesses a certain amount of residual pseudonym resource \(R_i^t\). Each pseudonym pool generates pseudonyms at a constant rate, \(\theta_i\). The average consuming rate of pseudonym resource of \(P_i\) in the following time (denoted as \(\lambda_i^t\)) can be estimated from the historical records by statistical methods. During time interval \(T_i\), if \(R_i^t > (\lambda_i^t - \theta_i)T\), \(P_i\) has a certain amount of redundant pseudonym resource. Otherwise, \(P_i\) lacks pseudonym resource. Let \(r_i^t\) represent the difference between the amount of required resources and the amount of actual resources as follows,

\[
r(P_i) = |R_i^t + \theta_iT - \lambda_i^tT|.
\]

\(P_i\) shares idle pseudonyms with other pseudonym pools or receives pseudonyms from others. We represent the pseudonym pool offering pseudonyms to others as OP, and the pseudonym pool receiving pseudonyms from the OPs as RP.

In an SDPS, a pseudonym resource scheduling problem includes three considerations.

- 1) OPs are rational to determine that how many idle pseudonyms can be offered to RPs after considering both the current and future demands.
- 2) To decrease the system overhead, OPs prefer to offer their idle pseudonyms to some proper RPs with smaller inter-pool communications cost.

According to this principle, an optimal pseudonym resource allocation strategy among the pseudonym pools can be designed.

VI. Solution for Pseudonym Resource Scheduling

A. The Optimal Strategies for OPs

For OPs, they offer a certain amount of idle pseudonyms to others according to a predefined utility function. The utility function of an OP, \(OP_i\), consists of two components: the satisfaction function and the cost function. The satisfaction function \(S_i^t\) is defined as

\[
S_i^t = w_i \log(1 + \rho_i x_i^t).
\]
Here, $x_i^t$ ($x_i^t \geq 0$) represents the amount of pseudonym resource that $OP_i$ would like to offer to others in time period $t$. $w_i$ is the willingness of $OP_i$, which is determined by its geographical advantage in $G$. $w_i$ can be expressed by

$$w_i = \frac{k}{\sum_{j \neq i} m(P_i, P_j)},$$

where $k$ is a predefined constant. The form of $w_i$ is similar to the closeness centrality in [35]. Clearly, less pseudonym transmission loss between OPs and other pseudonym pools stimulates $OP_i$ to share its idle pseudonyms. The redundant level in the current time period of $OP_i$ is denoted by

$$\rho_i = a \frac{R_i^t + \theta_i T}{\lambda_i T},$$

where $a$ is the redundant level gain and is predefined by the preference of pseudonym pools. $OP_i$ is willing to offer more pseudonyms to others for higher utility, when it possesses more idle pseudonyms. But $OP_i$ should take its demand level of the next time period (denoted as $\gamma_i^t$) into consideration when offering idle pseudonyms to others. $\gamma_i^t$ is defined as

$$\gamma_i^t = b \frac{\lambda_i^{t+1}}{\lambda_i^t},$$

where $b$ is the redundant level gain, that can be predefined. The cost of $OP_i$ offering resources to others is proportional to $\gamma_i^t$. Thus, the utility function of $OP_i$ can be expressed as

$$u_i^t = w_i \log(1 + \rho_i x_i^t) - \gamma_i^t x_i^t.$$

Next, to obtain the optimal solution, we analyze the characteristic of the utility function. Differentiating $u_i^t$ with respect to $x_i^t$, we get

$$\frac{\partial u_i^t}{\partial x_i^t} = \frac{w_i \rho_i x_i^t}{1 + \rho_i x_i^t} \ln 2 - \gamma_i^t,$$

$$\frac{\partial^2 u_i^t}{\partial x_i^t^2} = \frac{w_i \rho_i x_i^t}{(1 + \rho_i x_i^t)^2} \ln 2 < 0.$$

The utility function is concave, so we can obtain its maximal value by leveraging $\frac{\partial u_i^t}{\partial x_i^t} = 0$. Thus, the optimal amount of idle pseudonyms offering to others (denoted as $x_i^{t*}$) is expressed as

$$x_i^{t*} = \frac{w_i}{\gamma_i^t \ln 2} - \frac{1}{\rho_i^t}.$$

For the sake of fairness, $x_i^{t*}$ is constrained by $r(OP_i)$ as follows,

$$x_i^{t*} = \min(r(OP_i), \frac{w_i}{\gamma_i^t \ln 2} - \frac{1}{\rho_i^t}).$$

### B. Two-sided Matching among the Pseudonym Pools

After calculating the optimal number of idle pseudonyms provided by the OPs, a global controller in the central cloud decides how to allocate these pseudonyms to the RPs. The OPs transfer their idle pseudonyms to appropriate RPs for less cost of the inter-pool communications. It is a matching problem between the RPs and the OPs to decide that how to match an optimal RP for every OP, which aims at decreasing the system overhead resulted from inter-pool communications.

We use a simple and efficient two-sided matching theory based on Gale-Shapley algorithm to solve the problem of optimal pseudonym resource allocation [36]. RPs, as the invitees, will propose to the invitees OPs according to their own preference lists (denoted as $PL(P_i)$). The $PL$ is generated and stored according to communication cost of different pseudonym pools. In the preference list of $RP_i$, $OP_j$ is arranged in the $\phi_j^t$th order. Conversely, in the preference list of $OP_j$, $RP_i$ is arranged in the $\phi_i^t$th order. The preference lists are described as follows:

$$OP_j = PL(RP_i, \phi_j^t),$$

$$RP_i = PL(OP_j, \phi_i^t).$$

We take a pseudonym pool network consisting of two OPs and three RPs as an example. The preference lists of OPs and RPs are given as follows.

$$OP_1 : \{RP_2, RP_3, RP_2\};$$

$$OP_2 : \{RP_2, RP_1, RP_1\};$$

$$RP_1 : \{OP_2, OP_1\};$$

$$RP_2 : \{OP_2, OP_2\};$$

$$RP_3 : \{OP_1, OP_2\};$$

For simplicity, we consider that every RP demands the equal amount of pseudonyms and the redundant pseudonym resource of every OP only can satisfy one RP. In the first round of matching procedure, every RP proposes to its favorite OP according to its preference list. In the first round of result, every OP chooses the favorite one from the existing invitors according to the preference list. More details are shown as follows:

<table>
<thead>
<tr>
<th>1st round procedure</th>
<th>1st round result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$RP_1 \rightarrow OP_1$</td>
<td>$RP_1 \rightarrow$</td>
</tr>
<tr>
<td>$RP_2 \rightarrow OP_2$</td>
<td>$RP_2 \leftrightarrow OP_2$</td>
</tr>
<tr>
<td>$RP_3 \rightarrow OP_1$</td>
<td>$RP_3 \leftrightarrow OP_1$</td>
</tr>
</tbody>
</table>

$OP_1$ chooses to match with $RP_3$ temporarily because that $RP_3$ is the only inviter for $OP_1$ in the first round. $OP_2$ chooses to match with $RP_2$ because that $RP_2$ is prior to $RP_1$ in the preference list of $OP_2$. Then $RP_2$ has to choose the next OP in its preference list in the next round. Similarly, the second round procedure and result are listed as

<table>
<thead>
<tr>
<th>2nd round procedure</th>
<th>2nd round result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$RP_1 \rightarrow OP_1$</td>
<td>$RP_1 \leftrightarrow OP_1$</td>
</tr>
<tr>
<td>$RP_2 \leftrightarrow OP_2$</td>
<td>$RP_2 \leftrightarrow OP_2$</td>
</tr>
<tr>
<td>$RP_3 \leftrightarrow OP_1$</td>
<td>$RP_3 \rightarrow$</td>
</tr>
</tbody>
</table>

After being rejected by $OP_2$, $RP_1$ proposes to $OP_1$ in the second round. Due to the priority of $RP_1$, $OP_1$ prefers to break the previous matching result with $RP_3$, and then receives the invitation from $RP_1$. As a result, $RP_3$ has to stay alone in this round. Although $RP_3$ tries to propose $OP_2$ subsequently, the result in the second round is stable because that both $OP_1$ and $OP_2$ do not want to change their current invitors. Thus, two stable matches between $RP_1$, $RP_2$, $RP_3$ and $OP_1$, $OP_2$ are formed and satisfy the requirement of the two-sided matching. According to the above example, we know that, to decrease the communication cost between the pseudonym pools, the
matching problem between RPs and OPs can be solved by a two-sided matching problem.

We use a binary variable, \( \mu(RP_i, OP_j) \), to denote the final matching result. When the binary value is 1, it means that the pseudonym pools are matched. There may exist many rounds during the process of two-sided matching. Every matching round includes the following three stages.

1) Stage 1: The inviters propose to the invitees. RPs request pseudonym resource and send queries to the first OP in their preference lists. Every OP that act as the invitee selects the best partner according to its own preference list. When multiple RPs propose to the same OP, the OP selects the best RP from the proposers. If an RP is rejected by any OP, the RP will propose to the next OP in the RP’s preference list until it is accepted or is rejected by all the OPs in its preference list.

**Theorem 1:** \( \mu(RP_i, OP_j) = 1 \) will exist if and only if

\[
\sum_{s=1}^{\phi} \mu(RP_i, PL(RP_s, s)) = \sum_{s=1}^{\phi} \mu(PL(OP_j, s), OP_j) = 0.
\]

**Proof:** \( RP_i \) proposes to \( OP_j \), which means that \( RP_i \) has already been rejected by those OPs that whose orders are prior to \( \phi \). The rejections are expressed by

\[
\sum_{s=1}^{\phi} \mu(RP_i, PL(RP_s, s)) = 0. \quad OP_j \text{ accepts } RP_i, \text{ only if } \mu(OP_j, s, OP_j) = 0. \quad \text{This means that for } RP_i, \text{ it has been rejected those OPs that are better than } OP_j \text{ in its preference list. So } OP_j \text{ is the best choice of } RP_i \text{ at that time. Conversely, for } OP_j, \text{ the acceptance of } RP_i \text{ is done because that there is no better inviter than } RP_i. \text{ Then, } \mu(RP_i, OP_j) = 1 \text{ will exist if and only if both } RP_i \text{ and } OP_j \text{ have been matched with their own best partner. In summary, the final outcome of matching is the optimal two-sided result, because both inviters and invitees have been matched with their own best partner. The matching result is stable since both the inviters and the invitees have no better choice [36].}

2) Stage 2: OPs decide the amount of transmitted pseudonym resource. If \( \mu(RP_i, OP_j) = 1 \), the amount of pseudonym resource transmission between \( RP_i \) and \( OP_j \) (denoted as \( t(RP_i, OP_j) \)) depends on \( m(RP_i, OP_j) \), \( x(OP_j) \) and \( r(RP_i) \). For decreasing transmission cost, the amount of transmitted pseudonym resource is given by

\[
t(RP_i, OP_j) = \begin{cases} 
  r(RP_i) + m(RP_i, OP_j), & r(RP_i) + m(RP_i, OP_j) < x(OP_j); \\
  x(OP_j), & m(RP_i, OP_j) < x(OP_j) \leq r(RP_i) + m(RP_i, OP_j); \\
  0, & x(OP_j) \leq m(RP_i, OP_j).
\end{cases}
\]

(10)

The actual amount of pseudonym resource received by \( RP_i \) is equal to \( \min(t(RP_i, OP_j) - m(RP_i, OP_j), 0) \).

3) Stage 3: Updating the members of inviters and invitees. If \( \mu(RP_i, OP_j) = 1 \) and \( RP_i \) obtains enough pseudonym resource, which satisfies \( \min(t(RP_i, OP_j) - m(RP_i, OP_j), 0) = r(RP_i) \), \( RP_i \) will split from the set of RPs. Otherwise, \( RP_i \) will update its resource status information as follows,

\[
r(RP_i) = r(RP_i) - \min(t(RP_i, OP_j) - m(RP_i, OP_j), 0),
\]

and then joins into the next matching round. Thus, a new set of RPs occurs. \( OP_j \) will update the status information after offering pseudonym resource to \( RP_i \) as

\[
r(OP_j) = r(OP_j) - t(RP_i, OP_j).
\]

(12)

If \( OP_j \) cannot offer enough amount of pseudonyms for any \( RP_i \) in the next round, which satisfies \( x(OP_j) \leq \min(m(RP_i, OP_j)) \), it will split from the set of OPs. Otherwise, it still stay in OPs. When the set of RPs or OPs is empty, the matching process ends.

### C. Pseudonym-flow Table

The optimal pseudonyms allocation strategy can be performed in terms of designing a detailed pseudonym-flow table for every OpenFlow switch. For a local cloud, it first satisfies the local pseudonym demands and then transfers redundant pseudonyms to others. The local clouds transfer pseudonyms to local vehicles or other local clouds in a batch. For instance, several pseudonyms are generated in \( OP_j \) and packaged together in time slot \( t \). We denote this pseudonym package as \( p^t \). According to the optimal pseudonym resource allocation strategy, \( OP_i \) should transfer \( t_{i,j} \) (the number of pseudonym packages) to \( RP_j \) (\( j = 1, 2, 3...N \)). If there exists a local pseudonym request at this time, \( p^t \) will be delivered to the local requester otherwise it will be transferred to \( RP_i \) or be stored in local pseudonym pool when \( \sum t_{i,j} = 0 \). Following this principle, a detailed pseudonym-flow table of \( OP_i \) can be designed according to Algorithm 1.

### VII. Numerical Results

In this section, we evaluate the performance of the proposed pseudonym resource scheme in an actual urban area of San Francisco. The latitude is from 37.73619 to 37.81505, and the longitude is from -122.51431 to -122.36731. As shown in Fig. 4, the observed area is approximately 11.03 x 7.6 km\(^2\), which is divided into 8 grids (local clouds) according to the spatial distribution of vehicle hotspots in Fig. 5 [37]. The coverage of each local cloud is about 11 km\(^2\). In an urban area, the vehicles often take familiar routes in a specified time period, such as similar trajectories from home to work in the day time [38]. We also deploy 8 pseudonym pools in the observed area shown in Fig. 4, whose locations are restricted by the geographical conditions and the traffic load of each local cloud. The pseudonym pools 1, 2, 3 and 4 are deployed in the commercial areas. And the pseudonym pools 5, 6, 7 and 8 belong to the residential areas. This deployment strategy of local clouds follows the spatio-temporal distributions of the vehicles.

In this paper, we use the OpenFlow protocol to deploy the SDN [16]. Every pseudonym pool connects with an OpenFlow switch, which is responsible for forwarding the pseudonym flow. A global OpenFlow controller is deployed at a remote cloud, which acts as the central cloud. There exists a data
Algorithm 1 Pseudonym distribution algorithm


1: Initialize an array $A[N] = 0$ and $j = 0$.
2: while $t \in T$ do
3:   Generate a pseudonym package, $p^i_t$.
4:   if there is a local pseudonym request then
5:     Deliver $p^i_t$ to the local requester.
6:   else
7:     Initialize Flag ← 0.
8:     while $\sum_{k=0}^{N-1} A[k] < \sum_{k=1}^{N} t_{i,k}$ AND Flag do
9:       while $j < N$ AND Flag do
10:          if $A[j] < t_{i,j+1}$ then
11:            Transfer $p^i_t$ to $RP_j+1$.
13:            Next RP, $j ← j + 1$
14:          end if
15:       end while
16:     end while
17:     if $\sum_{k=0}^{N-1} A[k] = \sum_{k=1}^{N} t_{i,k}$ then
18:       Store $p^i_t$ locally.
19:     end if
20: end while
21: Next time slot, $t ← t + 1$
22: end while

Fig. 4: The connection of pseudonym pools in the real map of San Francisco.

Fig. 5: Spatial distribution of vehicle hotspots.

center in the central cloud, which collects global real-time status information of the network. The OpenFlow controller can access to the global information for predictions, analyses and decisions. According to pseudonym-flow rules designed by the OpenFlow controller, pseudonym resource is scheduled among the pseudonym pools.

The pseudonym pools communicate with each other through wired communication technologies. For simplicity, the transmission cost of pseudonyms (i.e., package dropout rate) is set to 1 unit/km and the generating rate of each pseudonym pool can be equal, denoted by $\theta$ [14]. According to the vehicular statistic data in [37], we set that the pseudonyms consuming rate of each pseudonym pool follows the interval [100, 400] units per minute with an observation time period (i.e., 1 hour).

Actually, the wired connections among the pseudonym pools are restricted by geographical conditions. For example, pseudonym pool 7 is screened on three sides by three hills. It cannot directly connect to pseudonym pool 4, 6 and 8 since there are some geographical obstructions, such as hills and lakes. The network construction expense is too high to establish communication links across the hills. Therefore pseudonym pool 7 only establishes network connectivity with pseudonym pool 1 to decrease the network construction expense. The nearby pseudonym pools without geographical obstructions are directly connected with each other. Some pseudonym pools, that are far away from others, can also connect with each other through multi-hop transmission, e.g., pseudonym pools 2 and 6.

A. Performance Comparison of Different Approaches

To further analyze the performance of our proposed approach, we consider a typical scenario of unbalanced demands of pseudonyms among pseudonym pools. The pseudonym pools are divided into two sides: four pseudonym pools receiving pseudonyms from OPs (denoted as $RP = \{P_1, P_2, P_3, P_4\}$) and four pseudonym pools that offer pseudonyms to others (denoted as $OP = \{P_5, P_6, P_7, P_8\}$). The pseudonym generating rate of each pseudonym pool is 100 units/minute. During the observation period of 1 hour, if the pseudonyms consuming rates of the RPs range from 300 to 400 units/minute, it indicates that the RPs are busy. While if pseudonym consuming rates of the OPs range from 100 to 200 units/minute, it indicates that the RPs are idle.

The pseudonym pools cooperate to share idle pseudonyms using two-sided matching theory. During the observation time,
the probability of vehicles obtaining pseudonyms from a pseudonym pool is expressed by $PO_i = \min(1 - \frac{\theta_i T}{N}, 1)$. Fig. 6 shows the performance comparison of different approaches with respect to the probability of vehicles obtaining pseudonyms from pseudonym pools. From this figure, OPs are always able to satisfy the demands of the vehicles during observation time. It means that idle pseudonym resource sharing has no influence on the performance of OPs since the OPs have enough pseudonyms to satisfy pseudonym demands of local vehicles. Moreover, RPs are able to improve $PO_i$ through obtaining some pseudonyms from OPs with the help of pseudonym-sharing. As a result, the performance of the whole network is improved when the pseudonym pools cooperate with each other. The average value of probability in our proposed approach is about 28% higher than that without pseudonym-cooperation. Similar improvement can be observed in Fig. 7, where the performance index is the total number of served vehicles. The average value of the total number of served vehicles with our proposed scheme is 40% higher than that without cooperation scheme.

Generally, the level of location privacy is quantified as the uncertainty of the information related to a specific vehicle. Here, the uncertainty is described by privacy entropy $H$. The maximum $H$ of a local cloud is given by $H_{m} = \log_2(|S|)$ [39]. Here, $|S|$ represents the total number of served vehicles, which can obtain the needed pseudonyms. Fig. 8 shows that the improvement of the average entropy of vehicles with cooperation is influenced by the pseudonyms generating rate $\theta$. When the generating rate of pseudonyms is 50 units/minute, the maximum improved entropy is 12% in our scheme. This emphasizes the importance of pseudonym-cooperation among pseudonym pools to improve the privacy entropy of vehicles when the pseudonym-generating rate is low. As the generating rate of the pseudonym pools increases, the change of average entropy is not obvious. It is because most of the pseudonym pools can gradually satisfy the vehicles’ demands by them-
Fig. 9 shows system overhead comparison between our proposed scheme with the existing scheme [14]. One of the existing schemes only schedules pseudonym resource among nearby local clouds, which is called as Nearby Pseudonym Pools Matching (NPPM) scheme in this paper. Fig. 9 shows that our proposed scheme has less system overhead than that of the NPPM scheme. It is because that pseudonym resource in our scheme is scheduled via a global optimal way. In our scheme, the two-sided matching theory is utilized to decreases system overhead due to cost of inter-pool communication. While the NPPM scheme can only schedule a part of pseudonym resource among the nearby pseudonym pools, which generally takes more pseudonym-scheduling times to satisfy pseudonym demand. Especially, when $\theta$ decreases, the number of pseudonym-scheduling times is increasing leading to bigger system overhead.

**B. Impacts of Different System Parameters**

Fig. 10 shows the total amount of pseudonyms offered by OPs with respect to different system parameters. Here, we set the system parameters as $[k, a, b] = [100, 1, 0.5]$. The total amount of offered pseudonyms by OPs increases when the generating rate of pseudonyms $\theta$ increases. The figure shows that OPs can offer more idle pseudonyms when they generate more pseudonyms. The amount of offered pseudonyms is influenced by the following predefined parameters, $k$, $a$, $b$ and $\theta$. Fig. 10(a) shows the higher value of willingness constant ($k$) brings more pseudonyms offered by OPs, when other system parameters are fixed. Fig. 10(b) shows that when the redundant level constant ($a$) increases, the total amount of offered pseudonyms also increases. But Fig. 10(c) shows that the total amount of pseudonyms offered by OPs is decreased when OPs pay much more attention to predicted demand level ($b$) of the next time period. In summary, the system parameters, $\theta$, $k$ and $a$, are beneficial to increase the total amount of pseudonyms offered by OPs. While the pseudonym demands of the next time period brings negative influence to the total amount of offered pseudonyms. When the pseudonym-generation rate is high, the OPs are willing to share their idle pseudonyms to others. Otherwise, the OPs are not willing to share pseudonyms even if $k$, and $a$ are higher. Apparently, the OPs should first satisfy their own demand of pseudonyms, and then consider to help others.

**VIII. CONCLUSIONS**

In this paper, we have proposed a software defined pseudonym management scheme, which exploits the SDN technology to schedule and manage the pseudonyms among distributed pseudonym pools. We have designed a hierarchical architecture of SDPS for scheduling the pseudonym resource scheduling problem. To decrease the system overheads, due to the cost of inter-pool communication, we deployed a two-side matching theory to formulate and solve the problem. Through extensive numerical results, we have illustrated that our proposed approach is efficient in improving pseudonym-utilization, and that it also effectively strengthens the location privacy of the vehicles.

**IX. ACKNOWLEDGMENT**

The work is supported in part by programs of NSFC under Grant nos. 61422201, 61370159 and U1201253, U1301255,
REFERENCES


Xumin Huang is now a master student of networked control systems in Guangdong University of Technology, China. His research interests mainly focus on network performance analysis, simulation and enhancement in wireless communications and networking.

Rong Yu [S’05, M’08] received his Ph.D. degree from Tsinghua University, China, in 2007. He is a full professor Guangdong University of Technology (GDUT) now. His research interest mainly focuses on wireless communications and networking, including cognitive radio, wireless sensor networks, and home networking. He is the co-inventor of over 10 patents and author or co-author of over 70 international journal and conference papers. Dr. Yu is currently serving as the deputy secretary general of the Internet of Things (IoT) Industry Alliance, Guangdong, China, and the deputy head of the IoT Engineering Center, Guangdong, China. He is the member of home networking standard committee in China, where he leads the standardization work of three standards.

Jiawen Kang received the M.S. degree from the Guangdong University of Technology, China, in 2015. He is now pursuing his Ph.D. degree in Guangdong University of Technology, China. His research interests mainly focus on resource management, security and privacy protection in wireless communications and networking. He is the author or co-author of 15 papers published in journals, magazines, and proceedings of international conferences.

Ning Wang [M] received the M.E. degree in electronic engineering from Nanyang University, Singapore, and the Ph.D. degree in electronic engineering from the University of Surrey, Guildford, U.K., in 2000, and 2004, respectively. Since 2009, he has been the Principal Investigator for several EU and U.K. Research Grants in the areas of future internet design and network management and control. He is currently a Senior Lecturer with the Institute for Communication Systems, University of Surrey. His current research interests include information-centric networking, network resource management and optimization, and smart grid communications.

Sabita Maharjan [M’09] is currently a Post-Doctoral Fellow with the Simula Research Laboratory, Fornebu, Norway. She received M.E. degree from the Antenna and Propagation Laboratory, Tokyo Institute of Technology, Tokyo, Japan, in 2008, and Ph.D. degree in Networks and Distributed Systems the Simula Research Laboratory and University of Oslo, Oslo, Norway, in 2013. Her research interests include wireless networks, network security, smart grid communications, cyber-physical systems, and machine-to-machine communications.

Yan Zhang [SM’10] is currently Head of Department, Department of Networks at Simula Research Laboratory, Norway; and an Associate Professor (part-time) at the Department of Informatics, University of Oslo, Norway. He received a Ph.D. degree in School of Electrical & Electronics Engineering, Nanyang Technological University, Singapore. He is an associate editor or on the editorial board of a number of well-established scientific international journals, e.g., Wiley Wireless Communications and Mobile Computing (WCMC). He also serves as the guest editor for IEEE Transactions on Smart Grid, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Industrial Informatics, IEEE Communications Magazine, IEEE Wireless Communications, IEEE Network, IEEE Systems and IEEE Internet of Things. He serves as chair positions in a number of conferences, including IEEE PIMRC 2016, IEEE Cloudcom 2016/2015, IEEE CCNC 2016, IEEE ICC 2016, WICON 2016, and IEEE SmartGridComm 2015. He serves as TPC member for numerous international conference including IEEE INFOCOM, IEEE ICC, IEEE GLOBECOM, and IEEE WCNC. His current research interest include: wireless networks and reliable and secure cyber-physical systems (e.g., smart grid, transport, healthcare). He has received 8 Best Paper Awards. He is a senior member of IEEE, IEEE ComSoc, IEEE VT society, IEEE PES and IEEE Computer society. He is a Fellow of IET.